



Open Source Used In AppDynamics_Cloud_Common_Inge stion_Service 22.6.0

Cisco Systems, Inc.

www.cisco.com

Cisco has more than 200 offices worldwide.
Addresses, phone numbers, and fax numbers
are listed on the Cisco website at
www.cisco.com/go/offices.

Text Part Number: 78EE117C99-1340186480

This document contains licenses and notices for open source software used in this product. With respect to the free/open source software listed in this document, if you have any questions or wish to receive a copy of any source code to which you may be entitled under the applicable free/open source license(s) (such as the GNU Lesser/General Public License), please contact us at external-opensource-requests@cisco.com.

In your requests please include the following reference number **78EE117C99-1340186480**

Contents

1.1 fabric8--:kubernetes-model--:api-extensions 4.10.3

1.1.1 Available under license

1.2 apache-httpcomponents-core rel/v5.0.2

1.2.1 Available under license

1.3 json-simple 1.1.1

1.3.1 Available under license

1.4 junit-platform-junit-platform-commons 1.7.1

1.4.1 Available under license

1.5 apache-commons-configuration 1.8

1.5.1 Available under license

1.6 hk2-implementation-utilities 2.6.1

1.6.1 Available under license

1.7 reactive-streams v1.0.3

1.7.1 Available under license

1.8 kotlin-reflect 1.3.71

1.8.1 Available under license

1.9 pact-jvm-core-pact-broker 4.0.10

1.9.1 Available under license

1.10 jansi 1.17.1

1.10.1 Available under license

1.11 okio 2.8.0

1.11.1 Available under license

1.12 hdrhistogram 2.1.9

1.12.1 Available under license

1.13 micronaut 3.2.7

1.13.1 Available under license

- 1.14 micronaut-micrometer 4.2.0**
 - 1.14.1 Available under license
- 1.15 pact-jvm-consumer-java8 4.0.10**
 - 1.15.1 Available under license
- 1.16 appdynamics-java-agent-api 4.5.13.27526**
 - 1.16.1 Available under license
- 1.17 kafka-schema-serializer 5.5.1**
 - 1.17.1 Available under license
- 1.18 io-swagger-swagger-annotations 1.6.2**
 - 1.18.1 Available under license
- 1.19 elasticsearch-kubernetes-cloud-plugin 4.10.3**
 - 1.19.1 Available under license
- 1.20 hk2-api-module 2.6.1**
 - 1.20.1 Available under license
- 1.21 apache-kafka 3.0.0**
 - 1.21.1 Available under license
- 1.22 annotations-for-metrics 4.1.17**
 - 1.22.1 Available under license
- 1.23 fabric8-:::kubernetes-model-::-storage-class 4.10.3**
 - 1.23.1 Available under license
- 1.24 reactor-core 3.4.14**
 - 1.24.1 Available under license
- 1.25 jakarta-ws-rs-api 3.0.0**
 - 1.25.1 Available under license
- 1.26 wire-schema 3.2.2**
 - 1.26.1 Available under license
- 1.27 wire-protocol-buffer-runtime 3.6.0**
 - 1.27.1 Available under license
- 1.28 animal-sniffer-annotation 1.19**
 - 1.28.1 Available under license
- 1.29 checker-qual 3.12.0**
 - 1.29.1 Available under license
- 1.30 generex 1.0.2**
 - 1.30.1 Available under license
- 1.31 feign-jaxrs 8.18.0**
 - 1.31.1 Available under license
- 1.32 antlr-4-tool 4.7.2**
 - 1.32.1 Available under license
- 1.33 junit-5-bill-of-materials 5.7.1**

- 1.33.1 Available under license
- 1.34 apache-xml-commons 1.4.01**
 - 1.34.1 Available under license
- 1.35 opentracing-api 0.33.0**
 - 1.35.1 Available under license
- 1.36 bean-validation-api 2.0.1**
 - 1.36.1 Available under license
- 1.37 apache-commons-text 1.9**
 - 1.37.1 Available under license
- 1.38 nimbus-jose-jwt 9.15.2**
 - 1.38.1 Available under license
- 1.39 azure-java-sdk-bom-(bill-of-materials) 1.1.1**
 - 1.39.1 Available under license
- 1.40 fabric8-:::kubernetes-model-:::common 4.10.3**
 - 1.40.1 Available under license
- 1.41 jakarta-ws-rs-api 2.1.6**
 - 1.41.1 Available under license
- 1.42 okio 2.5.0**
 - 1.42.1 Available under license
- 1.43 commons-logging 1.1.1**
 - 1.43.1 Available under license
- 1.44 jackson-core 2.13.2**
 - 1.44.1 Available under license
- 1.45 jackson-xc 2.13.2**
 - 1.45.1 Available under license
- 1.46 fabric8-:::openshift-model 4.10.3**
 - 1.46.1 Available under license
- 1.47 jackson-module-parameter-names 2.13.2**
 - 1.47.1 Available under license
- 1.48 metrics-health-checks 4.1.17**
 - 1.48.1 Available under license
- 1.49 junit-platform-junit-platform-engine 1.7.2**
 - 1.49.1 Available under license
- 1.50 protocol-buffer-java-api 3.19.4**
 - 1.50.1 Available under license
- 1.51 pact-jvm-core-matchers 4.0.10**
 - 1.51.1 Available under license
- 1.52 google-guice 4.0**
 - 1.52.1 Available under license

- 1.53 google-android-annotations-library 4.1.1.4**
 - 1.53.1 Available under license
- 1.54 java-classmate classmate-1.5.1**
 - 1.54.1 Available under license
- 1.55 fabric8--kubernetes-model--metrics 4.10.3**
 - 1.55.1 Available under license
- 1.56 config 5.5.1**
 - 1.56.1 Available under license
- 1.57 apache-commons-lang 3.4**
 - 1.57.1 Available under license
- 1.58 aop-alliance 1.0**
 - 1.58.1 Available under license
- 1.59 junit-platform-junit-platform-engine 1.7.1**
 - 1.59.1 Available under license
- 1.60 jackson-datatype-jdk8 2.13.2**
 - 1.60.1 Available under license
- 1.61 commons-codec 1.13**
 - 1.61.1 Available under license
- 1.62 google-api-grpc-proto-google-common-protos 2.0.1**
 - 1.62.1 Available under license
- 1.63 wire-multiplatform-runtime-(experimental) 3.2.2**
 - 1.63.1 Available under license
- 1.64 bean-validation-api 2.0.2**
 - 1.64.1 Available under license
- 1.65 io-grpc-grpc-protobuf-lite 1.39.0**
 - 1.65.1 Available under license
- 1.66 dropwizard-lifecycle-support 2.0.18**
 - 1.66.1 Available under license
- 1.67 jul-to-slf4j-bridge 1.7.30**
 - 1.67.1 Available under license
- 1.68 metrics-integration-with-jmx 4.1.17**
 - 1.68.1 Available under license
- 1.69 jackson-module-afterburner 2.13.2**
 - 1.69.1 Available under license
- 1.70 opentelemetry---io.opentelemetry:opentelemetry-proto 1.6.0-alpha**
 - 1.70.1 Available under license
- 1.71 wire-schema 3.6.0**
 - 1.71.1 Available under license
- 1.72 micronaut-kafka 4.2.0**

- 1.72.1 Available under license
- 1.73 fabric8--:kubernetes-model--:rbac 4.10.3**
 - 1.73.1 Available under license
- 1.74 jacoco v0.8.6**
 - 1.74.1 Available under license
- 1.75 micronaut-rxjava-3 2.2.1**
 - 1.75.1 Available under license
- 1.76 kotlinx-coroutines-io-jvm 0.1.16**
 - 1.76.1 Available under license
- 1.77 micronaut-coherence 3.0.1**
 - 1.77.1 Available under license
- 1.78 com.helger:profiler 1.1.1**
 - 1.78.1 Available under license
- 1.79 kafka-protobuf-serializer 5.5.1**
 - 1.79.1 Available under license
- 1.80 jackson-datatype-jsr310 2.13.2**
 - 1.80.1 Available under license
- 1.81 jakarta-annotations-api 2.0.0**
 - 1.81.1 Available under license
- 1.82 byte-buddy 1.12.7**
 - 1.82.1 Available under license
- 1.83 micronaut-oracle-cloud 2.1.1**
 - 1.83.1 Available under license
- 1.84 kafka-protobuf-provider 5.5.1**
 - 1.84.1 Available under license
- 1.85 junit-jupiter-junit-jupiter-engine 5.7.1**
 - 1.85.1 Available under license
- 1.86 reactive-relational-database-connectivity-bill-of-materials Arabba-SR11**
 - 1.86.1 Available under license
- 1.87 jboss-logging 3.3.2.Final**
 - 1.87.1 Available under license
- 1.88 feign-httpclient 8.18.0**
 - 1.88.1 Available under license
- 1.89 junit-platform-junit-platform-commons 1.8.2**
 - 1.89.1 Available under license
- 1.90 hamcrest 1.1**
 - 1.90.1 Available under license
- 1.91 caffeine-cache 2.9.1**
 - 1.91.1 Available under license

- 1.92 micronaut-reactor 2.2.2**
 - 1.92.1 Available under license
- 1.93 icu4j 70.1**
 - 1.93.1 Available under license
- 1.94 pact-jvm-provider 4.0.10**
 - 1.94.1 Available under license
- 1.95 jcip-annotations-under-apache-license 1.0-1**
 - 1.95.1 Available under license
- 1.96 abegocore 1.0.3**
 - 1.96.1 Available under license
- 1.97 zstd-jni 1.5.0-2**
 - 1.97.1 Available under license
- 1.98 apache-http-client 5.0.3**
 - 1.98.1 Available under license
- 1.99 opentracing-noop 0.33.0**
 - 1.99.1 Available under license
- 1.100 javax-ws-rs-api 2.1.1**
 - 1.100.1 Available under license
- 1.101 micrometer-bom 1.8.3**
 - 1.101.1 Available under license
- 1.102 apache-commons-collections 4.1**
 - 1.102.1 Available under license
- 1.103 metrics-utility-servlets 4.1.17**
 - 1.103.1 Available under license
- 1.104 java-diff-utils 1.3.0**
 - 1.104.1 Available under license
- 1.105 iterators 1.5**
 - 1.105.1 Available under license
- 1.106 atomicfu-common 0.14.1**
 - 1.106.1 Available under license
- 1.107 io-swagger-swagger-annotations 1.6.0**
 - 1.107.1 Available under license
- 1.108 error_prone_annotations 2.7.1**
 - 1.108.1 Available under license
- 1.109 servicelocator-default-implementation 2.6.1**
 - 1.109.1 Available under license
- 1.110 jackson-jaxrs 2.13.2**
 - 1.110.1 Available under license
- 1.111 arrow-type-classes 0.9.0**

- 1.111.1 Available under license
- 1.112 atomicfu 0.14.1**
 - 1.112.1 Available under license
- 1.113 io-grpc-grpc-context 1.39.0**
 - 1.113.1 Available under license
- 1.114 hibernate-validator 6.1.7.Final**
 - 1.114.1 Available under license
- 1.115 pact-jvm-consumer-junit5 4.0.10**
 - 1.115.1 Available under license
- 1.116 joda-time v2.10.9**
 - 1.116.1 Available under license
- 1.117 okio 2.5.0**
 - 1.117.1 Available under license
- 1.118 fabric8--:kubernetes-model--:certificates 4.10.3**
 - 1.118.1 Available under license
- 1.119 protocol-buffer-java-util-package 3.17.2**
 - 1.119.1 Available under license
- 1.120 jetbrains-kotlin-kotlin-stdlib-jdk8 1.3.71**
 - 1.120.1 Available under license
- 1.121 opentest4j-opentest4j 1.2.0**
 - 1.121.1 Available under license
- 1.122 fabric8--:kubernetes-model--:scheduling 4.10.3**
 - 1.122.1 Available under license
- 1.123 apache-groovy 3.0.9**
 - 1.123.1 Available under license
- 1.124 kotson 2.5.0**
 - 1.124.1 Available under license
- 1.125 ktor-server-core 1.2.6**
 - 1.125.1 Available under license
- 1.126 commons-io 2.11.0**
 - 1.126.1 Available under license
- 1.127 mockito-junit-jupiter 4.3.1**
 - 1.127.1 Available under license
- 1.128 jersey-inject-hk2 2.32**
 - 1.128.1 Available under license
- 1.129 reflections 0.9.10**
 - 1.129.1 Available under license
- 1.130 jersey-ext-bean-validation 2.32**
 - 1.130.1 Available under license

- 1.131 asm-commons 8.0.1**
 - 1.131.1 Available under license
- 1.132 evo-inflector 1.2.2**
 - 1.132.1 Available under license
- 1.133 archaius-core 0.7.6**
 - 1.133.1 Available under license
- 1.134 micronaut-security 3.5.0**
 - 1.134.1 Available under license
- 1.135 fabric8-::kubernetes-model-::networking 4.10.3**
 - 1.135.1 Available under license
- 1.136 micronaut-tracing 4.1.1**
 - 1.136.1 Available under license
- 1.137 wire-protocol-buffer-runtime 3.2.2**
 - 1.137.1 Available under license
- 1.138 dropwizard 2.0.18**
 - 1.138.1 Available under license
- 1.139 mockito 4.3.1**
 - 1.139.1 Available under license
- 1.140 alpn-api 1.1.3.v20160715**
 - 1.140.1 Available under license
- 1.141 slf4j-api-module 1.7.29**
 - 1.141.1 Available under license
- 1.142 jcl-1-2-implemented-over-slf4j 1.7.30**
 - 1.142.1 Available under license
- 1.143 dropwizard-jersey-support 2.0.18**
 - 1.143.1 Available under license
- 1.144 apache-httmime 4.5.5**
 - 1.144.1 Available under license
- 1.145 config-com-typesafe-config 1.3.1**
 - 1.145.1 Available under license
- 1.146 micronaut-email 1.2.0**
 - 1.146.1 Available under license
- 1.147 jakarta-expression-language-3.0-implementation 4.0.2**
 - 1.147.1 Available under license
- 1.148 dropwizard-servlet-support 2.0.18**
 - 1.148.1 Available under license
- 1.149 fabric8-::kubernetes-model-::policy 4.10.3**
 - 1.149.1 Available under license
- 1.150 apache-xerces2-j 2.12.2**

- 1.150.1 Available under license
- 1.151 micronaut-data 3.3.0**
 - 1.151.1 Available under license
- 1.152 pact-jvm-consumer 4.0.10**
 - 1.152.1 Available under license
- 1.153 micronaut-serialization 1.0.1**
 - 1.153.1 Available under license
- 1.154 apache-groovy 3.0.9**
 - 1.154.1 Available under license
- 1.155 ktor-http-cio 1.2.6**
 - 1.155.1 Available under license
- 1.156 dropwizard-validation-support 2.0.18**
 - 1.156.1 Available under license
- 1.157 ktor-utils-jvm 1.2.6**
 - 1.157.1 Available under license
- 1.158 opentelemetry-java---io.opentelemetry:opentelemetry-sdk-metrics 1.6.0-alpha**
 - 1.158.1 Available under license
- 1.159 lz4-and-xxhash 1.7.1**
 - 1.159.1 Available under license
- 1.160 fabric8--:-kubernetes-model--:-autoscaling 4.10.3**
 - 1.160.1 Available under license
- 1.161 javassist 3.27.0**
 - 1.161.1 Available under license
- 1.162 kafka-avro-serializer 5.5.1**
 - 1.162.1 Available under license
- 1.163 protocol-buffers-[bom] 3.17.2**
 - 1.163.1 Available under license
- 1.164 jetbrains-kotlin-kotlin-stdlib-jdk7 1.3.71**
 - 1.164.1 Available under license
- 1.165 micronaut-mongodb 4.1.0**
 - 1.165.1 Available under license
- 1.166 io-grpc-grpc-protobuf 1.39.0**
 - 1.166.1 Available under license
- 1.167 junit-jupiter-junit-jupiter-api 5.7.1**
 - 1.167.1 Available under license
- 1.168 fabric8--:-kubernetes-model--:-apps 4.10.3**
 - 1.168.1 Available under license
- 1.169 utils 6.2.0**

- 1.169.1 Available under license
- 1.170 micronaut-problem-json 2.2.2**
 - 1.170.1 Available under license
- 1.171 jackson-datatype-joda 2.13.2**
 - 1.171.1 Available under license
- 1.172 kotlin-stdlib 1.6.10**
 - 1.172.1 Available under license
- 1.173 apache-commons-lang 2.6**
 - 1.173.1 Available under license
- 1.174 rfc3986-uri 0.8**
 - 1.174.1 Available under license
- 1.175 metrics-jetty 4.1.17**
 - 1.175.1 Available under license
- 1.176 json-p 1.0.4**
 - 1.176.1 Available under license
- 1.177 kafka-protobuf-provider 6.2.0**
 - 1.177.1 Available under license
- 1.178 opentelemetry 1.6.0**
 - 1.178.1 Available under license
- 1.179 kafka-protobuf-serializer 6.2.0**
 - 1.179.1 Available under license
- 1.180 netty-project 4.1.74.Final**
 - 1.180.1 Available under license
- 1.181 jetbrains-kotlin-kotlin-stdlib-jdk7 1.3.11**
 - 1.181.1 Available under license
- 1.182 dropwizard-request-logging-support 2.0.18**
 - 1.182.1 Available under license
- 1.183 kotlin-stdlib 1.3.71**
 - 1.183.1 Available under license
- 1.184 apache-httpcomponents-core 4.4.9**
 - 1.184.1 Available under license
- 1.185 opentracing-util 0.33.0**
 - 1.185.1 Available under license
- 1.186 antlr 4.7.2**
 - 1.186.1 Available under license
- 1.187 open-telemetry/opentelemetry-java 1.6.0**
 - 1.187.1 Available under license
- 1.188 micronaut-kubernetes 3.4.0**
 - 1.188.1 Available under license

- 1.189 java-servlet-api 4.0.4**
 - 1.189.1 Available under license
- 1.190 osgi-resource-locator-bundle 1.0.3**
 - 1.190.1 Available under license
- 1.191 ktor-network 1.2.6**
 - 1.191.1 Available under license
- 1.192 io.grpc:grpc-bom 1.39.0**
 - 1.192.1 Available under license
- 1.193 micronaut 3.4.1**
 - 1.193.1 Available under license
- 1.194 apache-http-client 4.5.13**
 - 1.194.1 Available under license
- 1.195 byte-buddy-agent 1.12.7**
 - 1.195.1 Available under license
- 1.196 fabric8::-kubernetes-model::-core 4.10.3**
 - 1.196.1 Available under license
- 1.197 google-guice 4.1.0**
 - 1.197.1 Available under license
- 1.198 kotlinx-coroutines-core 1.3.2-1.3.60**
 - 1.198.1 Available under license
- 1.199 fabric8::-kubernetes-model::-settings 4.10.3**
 - 1.199.1 Available under license
- 1.200 jackson-dataformat-yaml 2.13.2**
 - 1.200.1 Available under license
- 1.201 jackson-databind 2.13.2.2**
 - 1.201.1 Available under license
- 1.202 micronaut-gcp 4.1.1**
 - 1.202.1 Available under license
- 1.203 classgraph 4.8.25**
 - 1.203.1 Available under license
- 1.204 arrow-core-data 0.9.0**
 - 1.204.1 Available under license
- 1.205 bom 4.1.74.Final**
 - 1.205.1 Available under license
- 1.206 objenesis 3.2**
 - 1.206.1 Available under license
- 1.207 micronaut-r2dbc 2.1.0**
 - 1.207.1 Available under license
- 1.208 commons-logging 1.2**

- 1.208.1 Available under license
- 1.209 ktor-server-host-common 1.2.6**
 - 1.209.1 Available under license
- 1.210 okhttp 3.12.12**
 - 1.210.1 Available under license
- 1.211 jsr311-api 1.1.1**
 - 1.211.1 Available under license
- 1.212 javabeans-activation-framework-api 1.2.2**
 - 1.212.1 Available under license
- 1.213 arrow-annotations 0.9.0**
 - 1.213.1 Available under license
- 1.214 argparse 0.8.1**
 - 1.214.1 Available under license
- 1.215 opentelemetry-java 1.6.0**
 - 1.215.1 Available under license
- 1.216 jetbrains-kotlin-kotlin-stdlib-jdk8 1.6.10**
 - 1.216.1 Available under license
- 1.217 kotlinx-io-jvm 0.1.16**
 - 1.217.1 Available under license
- 1.218 jakarta-activation 1.1**
 - 1.218.1 Available under license
- 1.219 fabric8-::-kubernetes-model-::-discovery 4.10.3**
 - 1.219.1 Available under license
- 1.220 micronaut-azure 3.2.3**
 - 1.220.1 Available under license
- 1.221 dropwizard-logging-support 2.0.18**
 - 1.221.1 Available under license
- 1.222 jvm-integration-for-metrics 4.1.17**
 - 1.222.1 Available under license
- 1.223 stringtemplate4 4.1**
 - 1.223.1 Available under license
- 1.224 kotlin-stdlib-common 1.6.10**
 - 1.224.1 Available under license
- 1.225 dmfs/optional 0.3**
 - 1.225.1 Available under license
- 1.226 fabric8-::-kubernetes-model-::-events 4.10.3**
 - 1.226.1 Available under license
- 1.227 micronaut-servlet 3.2.2**
 - 1.227.1 Available under license

- 1.228 netty/transport/classes/kqueue 4.1.74.Final**
 - 1.228.1 Available under license
- 1.229 junit 4.10**
 - 1.229.1 Available under license
- 1.230 kotlinx-coroutines-jdk 1.3.2-1.3.60**
 - 1.230.1 Available under license
- 1.231 commons-compress 1.21**
 - 1.231.1 Available under license
- 1.232 io-grpc-grpc-stub 1.39.0**
 - 1.232.1 Available under license
- 1.233 jersey-ext-entity-filtering 2.31**
 - 1.233.1 Available under license
- 1.234 micronaut-test 3.1.1**
 - 1.234.1 Available under license
- 1.235 annotation-for-jdk-requirements 1.0**
 - 1.235.1 Available under license
- 1.236 json-java 20160212**
 - 1.236.1 Available under license
- 1.237 apache-commons-lang 3.11**
 - 1.237.1 Available under license
- 1.238 junit-jupiter-junit-jupiter-api 5.8.2**
 - 1.238.1 Available under license
- 1.239 snappy-java 1.1.8.1**
 - 1.239.1 Available under license
- 1.240 jetty-java-based-http-1-x-http-2-servlet-websocket-server 11.0.6**
 - 1.240.1 Available under license
- 1.241 snake-yaml 1.30**
 - 1.241.1 Available under license
- 1.242 protobuf-java-format 1.2**
 - 1.242.1 Available under license
- 1.243 jersey's-jersey 3.0.2**
 - 1.243.1 Available under license
- 1.244 kotlinx-coroutines-bom 1.5.1**
 - 1.244.1 Available under license
- 1.245 jetbrains-kotlin-kotlin-stdlib-jdk7 1.6.10**
 - 1.245.1 Available under license
- 1.246 javax-annotation-api 1.3.2**
 - 1.246.1 Available under license
- 1.247 kqueue 4.1.74.Final**

- 1.247.1 Available under license
- 1.248 ktor-network-tls 1.2.6**
 - 1.248.1 Available under license
- 1.249 feign-slf4j 8.18.0**
 - 1.249.1 Available under license
- 1.250 perfmark-perfmark-api 0.23.0**
 - 1.250.1 Available under license
- 1.251 jakarta-dependency-injection 2.0.1**
 - 1.251.1 Available under license
- 1.252 dropwizard-jetty-support 2.0.18**
 - 1.252.1 Available under license
- 1.253 throttling-appender 1.1.0**
 - 1.253.1 Available under license
- 1.254 kotlinox-io 0.1.16**
 - 1.254.1 Available under license
- 1.255 io-grpc-grpc-core 1.39.0**
 - 1.255.1 Available under license
- 1.256 netty/tomcatnative-[openssl---classes] 2.0.50.Final**
 - 1.256.1 Available under license
- 1.257 jackson-annotations 2.13.2**
 - 1.257.1 Available under license
- 1.258 junit-jupiter-junit-jupiter-engine 5.7.2**
 - 1.258.1 Available under license
- 1.259 jetbrains-annotations 13.0**
 - 1.259.1 Available under license
- 1.260 scala 2.12.8**
 - 1.260.1 Available under license
- 1.261 jersey-core-server 2.32**
 - 1.261.1 Available under license
- 1.262 ktor-bom 1.6.1**
 - 1.262.1 Available under license
- 1.263 micronaut-groovy 3.1.0**
 - 1.263.1 Available under license
- 1.264 io-swagger-swagger-annotations 2.1.12**
 - 1.264.1 Available under license
- 1.265 kafka-schema-registry-client 6.2.0**
 - 1.265.1 Available under license
- 1.266 asm 8.0.1**
 - 1.266.1 Available under license

1.267 dropwizard-jackson-support 2.0.18

1.267.1 Available under license

1.268 jackson-datatype-guava 2.13.2

1.268.1 Available under license

1.269 google-gson 2.8.6

1.269.1 Available under license

1.270 googleapis-core 0.0.3

1.270.1 Available under license

1.271 pact-jvm-core-model 4.0.10

1.271.1 Available under license

1.272 guava 31.0.1-jre

1.272.1 Available under license

1.273 asm-tree 8.0.1

1.273.1 Available under license

1.274 micronaut-rxjava-2 1.2.1

1.274.1 Available under license

1.275 jdbc3-bom 3.20.1

1.275.1 Available under license

1.276 netty-transport-native-unix-common 4.1.74.Final

1.276.1 Available under license

1.277 testcontainers-bom 1.16.2

1.277.1 Available under license

1.278 fabric8-::-kubernetes-model-::-batch 4.10.3

1.278.1 Available under license

1.279 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava

1.279.1 Available under license

1.280 fabric8-::-kubernetes-model-::-coordination 4.10.3

1.280.1 Available under license

1.281 kotlinox-coroutines-io 0.1.16

1.281.1 Available under license

1.282 cloudevents---kafka-transport-binding 2.1.1

1.282.1 Available under license

1.283 micronaut-grpc 3.1.3

1.283.1 Available under license

1.284 asm-analysis 8.0.1

1.284.1 Available under license

1.285 dropwizard-metrics-support 2.0.18

1.285.1 Available under license

1.286 jersey-ext-metainf-services 2.32

- 1.286.1 Available under license
- 1.287 io-grpc-grpc-netty 1.37.0**
 - 1.287.1 Available under license
- 1.288 arrow-core-extensions 0.9.0**
 - 1.288.1 Available under license
- 1.289 pact-jvm-core-support 4.0.10**
 - 1.289.1 Available under license
- 1.290 okhttp-logging-interceptor 3.12.12**
 - 1.290.1 Available under license
- 1.291 metrics-integration-for-jersey-2.x 4.1.17**
 - 1.291.1 Available under license
- 1.292 hystrix-core 1.5.18**
 - 1.292.1 Available under license
- 1.293 kotlin-logging 1.6.26**
 - 1.293.1 Available under license
- 1.294 findbugs-jsr305 3.0.2**
 - 1.294.1 Available under license
- 1.295 logback-core 1.2.11**
 - 1.295.1 Available under license
- 1.296 micronaut-sql-libraries 4.2.2**
 - 1.296.1 Available under license
- 1.297 pact-jvm-provider-junit5 4.0.10**
 - 1.297.1 Available under license
- 1.298 javax.inject:javax.inject 1**
 - 1.298.1 Available under license
- 1.299 micronaut-flyway 5.2.0**
 - 1.299.1 Available under license
- 1.300 feign-jackson 8.18.0**
 - 1.300.1 Available under license
- 1.301 kindedj 1.1.0**
 - 1.301.1 Available under license
- 1.302 micronaut-kotlin-integrations 3.2.2**
 - 1.302.1 Available under license
- 1.303 kotlin-stdlib-common 1.3.71**
 - 1.303.1 Available under license
- 1.304 cloudevents---api 2.1.1**
 - 1.304.1 Available under license
- 1.305 metrics-integration-for-logback 4.1.17**
 - 1.305.1 Available under license

- 1.306 micronaut-cache 3.3.0**
 - 1.306.1 Available under license
- 1.307 micronaut-aws 3.2.1**
 - 1.307.1 Available under license
- 1.308 log4j-over-slf4j 1.7.30**
 - 1.308.1 Available under license
- 1.309 rxjersey 0.9.0**
 - 1.309.1 Available under license
- 1.310 rxjava 2.2.21**
 - 1.310.1 Available under license
- 1.311 okio 2.8.0**
 - 1.311.1 Available under license
- 1.312 jersey-container-servlet-core 2.32**
 - 1.312.1 Available under license
- 1.313 feign-core 8.18.0**
 - 1.313.1 Available under license
- 1.314 commons-codec 1.15**
 - 1.314.1 Available under license
- 1.315 jackson-bom 2.13.2**
 - 1.315.1 Available under license
- 1.316 kotlin-libraries-bill-of-materials 1.6.10**
 - 1.316.1 Available under license
- 1.317 jackson-module:-guice 2.13.2**
 - 1.317.1 Available under license
- 1.318 apiguardian-apiguardian-api 1.1.2**
 - 1.318.1 Available under license
- 1.319 slf4j-api-module 1.7.25**
- 1.320 automation 1.11-8**
 - 1.320.1 Available under license
- 1.321 ktor-server-netty 1.2.6**
 - 1.321.1 Available under license
- 1.322 antlr 3.5.2**
 - 1.322.1 Available under license
- 1.323 java-semver 0.9.0**
 - 1.323.1 Available under license
- 1.324 spock-framework---bill-of-materials 2.0-groovy-3.0**
 - 1.324.1 Available under license
- 1.325 jersey-media-jaxb 2.32**
 - 1.325.1 Available under license

- 1.326 micronaut-mqtt 2.1.0**
 - 1.326.1 Available under license
- 1.327 jackson-integration-for-metrics 4.1.17**
 - 1.327.1 Available under license
- 1.328 kafka-protobuf-types 6.2.0**
 - 1.328.1 Available under license
- 1.329 kotlincoroutines-core-common 1.3.2-1.3.60**
 - 1.329.1 Available under license
- 1.330 fabric8::-kubernetes-model::-admission-registration,-authentication-and-authorization 4.10.3**
 - 1.330.1 Available under license
- 1.331 kafka-schema-registry-client 5.5.1**
 - 1.331.1 Available under license
- 1.332 wire-multiplatform-runtime-(experimental) 3.6.0**
 - 1.332.1 Available under license
- 1.333 java-architecture-for-xml-binding 2.3.3**
 - 1.333.1 Available under license
- 1.334 j2objc-annotations 1.3**
 - 1.334.1 Available under license
- 1.335 micronaut-redis 5.2.0**
 - 1.335.1 Available under license
- 1.336 apache-avro 1.11.0**
 - 1.336.1 Available under license
- 1.337 apache-httpcomponents-core-http/2 5.0.2**
 - 1.337.1 Available under license
- 1.338 rxjava 1.3.8**
 - 1.338.1 Available under license
- 1.339 jakarta-mail 1.5.0-b01**
 - 1.339.1 Available under license
- 1.340 fabric8::-kubernetes-model::-extensions 4.10.3**
 - 1.340.1 Available under license
- 1.341 kafka-schema-serializer 6.2.0**
 - 1.341.1 Available under license
- 1.342 zjsonpatch 0.3.0**
 - 1.342.1 Available under license
- 1.343 guava-internalfuturefailureaccess-and-internalfutures 1.0.1**
 - 1.343.1 Available under license
- 1.344 jetty-toolchain 1.0.4**
 - 1.344.1 Available under license

1.345 netty/transport/classes/epoll 4.1.74.Final

1.345.1 Available under license

1.346 ktor-http-jvm 1.2.6

1.346.1 Available under license

1.347 utils 5.5.1

1.347.1 Available under license

1.348 javax-inject1-as-osgi-bundle 2.6.1

1.348.1 Available under license

1.349 cloudevents---core 2.1.1

1.349.1 Available under license

1.350 metrics-core 4.1.17

1.350.1 Available under license

1.351 io-grpc-grpc-api 1.39.0

1.351.1 Available under license

1.352 apiguardian-apiguardian-api 1.1.0

1.352.1 Available under license

1.353 dropwizard-configuration-support 2.0.18

1.353.1 Available under license

1.354 junit-5-bill-of-materials 5.7.2

1.354.1 Available under license

1.355 slf4j-api-module 1.7.28

1.355.1 Available under license

1.356 jackson-jaxrs-base 2.13.2

1.356.1 Available under license

1.357 jersey's-jersey 2.34

1.357.1 Available under license

1.358 micronaut-views 3.2.0

1.358.1 Available under license

1.359 jersey-container-servlet 2.32

1.359.1 Available under license

1.360 dropwizard-utility-classes 2.0.18

1.360.1 Available under license

1.361 jersey-media-json-jackson 2.31

1.361.1 Available under license

1.1 fabric8-::kubernetes-model-::api-extensions 4.10.3

1.1.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.2 apache-httpcomponents-core rel/v5.0.2

1.2.1 Available under license :

Apache HttpComponents Core

Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation

source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.3 json-simple 1.1.1

1.3.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and
You must cause any modified files to carry prominent notices stating that You changed the files; and
You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works

as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.4 junit-platform-junit-platform-commons

1.7.1

1.4.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

* **a)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

* **b)** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive,

worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* ****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* ****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* ****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* ****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* ****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2)**** When the Program is Distributed as Source Code:

* ****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* ****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF

SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.5 apache-commons-configuration 1.8

1.5.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.6 hk2-implementation-utilities 2.6.1

1.6.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE
PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION
OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in

accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program,

the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS

SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient

receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price.

Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this

License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source

code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the

integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH

YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
'show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than 'show w' and 'show c'; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
program 'Gnomovision' (which makes passes at compilers) written by
James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is
making a combined work based on this library. Thus, the terms and
conditions of the GNU General Public License version 2 cover the whole
combination.

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent
modules, and to copy and distribute the resulting executable under
terms of your choice, provided that you also meet, for each linked
independent module, the terms and conditions of the license of that
module. An independent module is a module which is not derived from or
based on this library. If you modify this library, you may extend this
exception to your version of the library, but you are not obligated to
do so. If you do not wish to do so, delete this exception statement
from your version.

Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: <https://projects.eclipse.org/projects/ee4j.glassfish>

Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/glassfish-ha-api>
- * <https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor>
- * <https://github.com/eclipse-ee4j/glassfish-shoal>
- * <https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck>
- * <https://github.com/eclipse-ee4j/glassfish-jsftemplating>
- * <https://github.com/eclipse-ee4j/glassfish-hk2-extra>
- * <https://github.com/eclipse-ee4j/glassfish-hk2>
- * <https://github.com/eclipse-ee4j/glassfish-fighterfish>

Third-party Content

This project leverages the following third party content.

None

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.7 reactive-streams v1.0.3

1.7.1 Available under license :

Copyright Statement for Contributions to the Reactive Streams Project

=====

I hereby represent that all present, past and future contributions I make to the Reactive Streams project (which includes all repositories owned by the reactive-streams github organization) are governed by the Creative Commons Zero 1.0 Universal copyright statement, placing my contributions in the public domain. This entails that to the extent possible under law I waive all copyright and related or neighboring rights to the code or documents I contribute. I also represent that I have the authority to perform the above waiver with respect to the entirety of my contributions.

The text of the copyright statement is included in the COPYING file at the root of the reactive-streams repository at <https://github.com/reactive-streams/reactive-streams-jvm/blob/master/COPYING>.

Underwriting parties:

github name | Real Name, Email Address used for git commits, Company

-----+

rkuhn | Roland Kuhn, rk@rkuhn.info, Typesafe Inc.
benjchristensen | Ben Christensen, benjchristensen@gmail.com, Netflix Inc.
viktorklang | Viktor Klang, viktorklang@gmail.com, Typesafe Inc.
smaldini | Stephane Maldini, stephane.maldini@gmail.com, Pivotal Software Inc.
savulchik | Stanislav Savulchik, s.savulchik@gmail.com
ktoso | Konrad Malawski, konrad.malawski@project13.pl, Typesafe Inc.
ouertani | Slim Ouertani, ouertani@gmail.com
2m | Martynas Mickevicius, mmartynas@gmail.com, Typesafe Inc.
ldaley | Luke Daley, luke.daley@gradleware.com, Gradleware Inc.
colingodsey | Colin Godsey, crgodsey@gmail.com, MediaMath Inc.
davidmoten | Dave Moten, davidmoten@gmail.com
briantopping | Brian Topping, brian.topping@gmail.com, Mauswerks LLC
rstoyanchev | Rossen Stoyanchev, rstoyanchev@pivotal.io, Pivotal
BjornHamels | Bjorn Hamels, bjorn@hamels.nl
JakeWharton | Jake Wharton, jakewharton@gmail.com
anthonyvdotbe | Anthony Vanelverdinghe, anthonyv.be@outlook.com

seratch | Kazuhiro Sera, seratch@gmail.com, SmartNews, Inc.
akarnokd | David Karnok, akarnokd@gmail.com
egetman | Evgeniy Getman, getman.eugene@gmail.com
patriknw | Patrik Nordwall, patrik.nordwall@gmail.com, Lightbend Inc
angelsanz | ngel Sanz, angelsanz@users.noreply.github.com
shenghaiyang | , shenghaiyang@aliyun.com
kiiadi | Kyle Thomson, kylthoms@amazon.com, Amazon.com
jroper | James Roper, james@jazzy.id.au, Lightbend Inc.
olegdokuka | Oleh Dokuka, shadowgun@i.ua, Netifi Inc.
Scottmitch | Scott Mitchell, scott_mitchell@apple.com, Apple Inc.
retronym | Jason Zaugg, jzaugg@gmail.com, Lightbend Inc.
Creative Commons Legal Code

CC0 1.0 Universal

CREATIVE COMMONS CORPORATION IS NOT A LAW FIRM AND DOES NOT PROVIDE LEGAL SERVICES. DISTRIBUTION OF THIS DOCUMENT DOES NOT CREATE AN ATTORNEY-CLIENT RELATIONSHIP. CREATIVE COMMONS PROVIDES THIS INFORMATION ON AN "AS-IS" BASIS. CREATIVE COMMONS MAKES NO WARRANTIES REGARDING THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER, AND DISCLAIMS LIABILITY FOR DAMAGES RESULTING FROM THE USE OF THIS DOCUMENT OR THE INFORMATION OR WORKS PROVIDED HEREUNDER.

Statement of Purpose

The laws of most jurisdictions throughout the world automatically confer exclusive Copyright and Related Rights (defined below) upon the creator and subsequent owner(s) (each and all, an "owner") of an original work of authorship and/or a database (each, a "Work").

Certain owners wish to permanently relinquish those rights to a Work for the purpose of contributing to a commons of creative, cultural and scientific works ("Commons") that the public can reliably and without fear of later claims of infringement build upon, modify, incorporate in other works, reuse and redistribute as freely as possible in any form whatsoever and for any purposes, including without limitation commercial purposes. These owners may contribute to the Commons to promote the ideal of a free culture and the further production of creative, cultural and scientific works, or to gain reputation or greater distribution for their Work in part through the use and efforts of others.

For these and/or other purposes and motivations, and without any expectation of additional consideration or compensation, the person associating CC0 with a Work (the "Affirmer"), to the extent that he or she is an owner of Copyright and Related Rights in the Work, voluntarily elects to apply CC0 to the Work and publicly distribute the Work under its terms, with knowledge of his or her Copyright and Related Rights in the

Work and the meaning and intended legal effect of CC0 on those rights.

1. Copyright and Related Rights. A Work made available under CC0 may be protected by copyright and related or neighboring rights ("Copyright and Related Rights"). Copyright and Related Rights include, but are not limited to, the following:

- i. the right to reproduce, adapt, distribute, perform, display, communicate, and translate a Work;
- ii. moral rights retained by the original author(s) and/or performer(s);
- iii. publicity and privacy rights pertaining to a person's image or likeness depicted in a Work;
- iv. rights protecting against unfair competition in regards to a Work, subject to the limitations in paragraph 4(a), below;
- v. rights protecting the extraction, dissemination, use and reuse of data in a Work;
- vi. database rights (such as those arising under Directive 96/9/EC of the European Parliament and of the Council of 11 March 1996 on the legal protection of databases, and under any national implementation thereof, including any amended or successor version of such directive); and
- vii. other similar, equivalent or corresponding rights throughout the world based on applicable law or treaty, and any national implementations thereof.

2. Waiver. To the greatest extent permitted by, but not in contravention of, applicable law, Affirmer hereby overtly, fully, permanently, irrevocably and unconditionally waives, abandons, and surrenders all of Affirmer's Copyright and Related Rights and associated claims and causes of action, whether now known or unknown (including existing as well as future claims and causes of action), in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "Waiver"). Affirmer makes the Waiver for the benefit of each member of the public at large and to the detriment of Affirmer's heirs and successors, fully intending that such Waiver shall not be subject to revocation, rescission, cancellation, termination, or any other legal or equitable action to disrupt the quiet enjoyment of the Work by the public as contemplated by Affirmer's express Statement of Purpose.

3. Public License Fallback. Should any part of the Waiver for any reason be judged legally invalid or ineffective under applicable law, then the Waiver shall be preserved to the maximum extent permitted taking into account Affirmer's express Statement of Purpose. In addition, to the extent the Waiver is so judged Affirmer hereby grants to each affected person a royalty-free, non transferable, non sublicensable, non exclusive,

irrevocable and unconditional license to exercise Affirmer's Copyright and Related Rights in the Work (i) in all territories worldwide, (ii) for the maximum duration provided by applicable law or treaty (including future time extensions), (iii) in any current or future medium and for any number of copies, and (iv) for any purpose whatsoever, including without limitation commercial, advertising or promotional purposes (the "License"). The License shall be deemed effective as of the date CC0 was applied by Affirmer to the Work. Should any part of the License for any reason be judged legally invalid or ineffective under applicable law, such partial invalidity or ineffectiveness shall not invalidate the remainder of the License, and in such case Affirmer hereby affirms that he or she will not (i) exercise any of his or her remaining Copyright and Related Rights in the Work or (ii) assert any associated claims and causes of action with respect to the Work, in either case contrary to Affirmer's express Statement of Purpose.

4. Limitations and Disclaimers.

- a. No trademark or patent rights held by Affirmer are waived, abandoned, surrendered, licensed or otherwise affected by this document.
- b. Affirmer offers the Work as-is and makes no representations or warranties of any kind concerning the Work, express, implied, statutory or otherwise, including without limitation warranties of title, merchantability, fitness for a particular purpose, non infringement, or the absence of latent or other defects, accuracy, or the present or absence of errors, whether or not discoverable, all to the greatest extent permissible under applicable law.
- c. Affirmer disclaims responsibility for clearing rights of other persons that may apply to the Work or any use thereof, including without limitation any person's Copyright and Related Rights in the Work. Further, Affirmer disclaims responsibility for obtaining any necessary consents, permissions or other rights required for any use of the Work.
- d. Affirmer understands and acknowledges that Creative Commons is not a party to this document and has no duty or obligation with respect to this CC0 or use of the Work.

Licensed under Public Domain (CC0)

To the extent possible under law, the person who associated CC0 with this code has waived all copyright and related or neighboring rights to this code.

You should have received a copy of the CC0 legalcode along with this work. If not, see <http://creativecommons.org/publicdomain/zero/1.0/>.

1.8 kotlin-reflect 1.3.71

1.8.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2016 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/AbstractClassTypeConstructor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/structure/javaTypes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/JvmPackageScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/IncompatibleVersionErrorData.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/OverridingStrategy.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/Annotations.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/KotlinMetadataFinder.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/KClassifierImpl.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/structure/javaElements.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractTypeAliasDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/EmptyPackageFragmentDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ProtoBasedClassDataFinder.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
```

jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassifierDescriptor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/typeEnhancement.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/ErasedOverridabilityCondition.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/TypeIntersectionScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeProjectionBase.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/context.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/AbstractTypeConstructor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/full/exceptions.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractTypeParameterDescriptor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/FlexibleTypeDeserializer.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/predefinedEnhancementInfo.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/constants/IntegerValueTypeConstructor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/MetadataPackageFragmentProvider.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/full/KClassifiers.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/KotlinReflectionInternalError.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaTypeParameterDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/AbstractClassDescriptor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/RawType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AbstractDeserializedPackageFragmentProvider.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/TypeCheckingProcedure.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/typeQualifiers.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeSubstitutor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassifierDescriptorWithTypeParameters.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/KTypesJvm.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-

jar/kotlin/reflect/jvm/internal/KTypeParameterImpl.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeCapabilities.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/SuperCallReceiverValue.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/context.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/GivenFunctionsMemberScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/KotlinTypeCheckerImpl.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/TypeAliasDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaPropertyInitializerEvaluator.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ValueDescriptor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/types/JavaTypeResolver.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/MemberSignature.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeSubstitution.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaAnnotationDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/IntersectionTypeConstructor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/renderer/DescriptorRendererOptionsImpl.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/DeserializationComponentsForJava.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationDeserializer.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/flexibleTypes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/ClassicTypeCheckerContext.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/DescriptorEquivalenceForOverrides.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/KotlinTypeChecker.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/ErrorUtils.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/FunctionPlaceholders.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/renderer/ClassifierNamePolicy.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-

jar/kotlin/reflect/jvm/internal/impl/Utils/SmartList.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedTypeParameterDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JavaFlexibleTypeDeserializer.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/full/KTypes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/Utils.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassConstructorDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/Utils/DFS.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/calls/inference/CapturedTypeConstructor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/types/RawType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/dynamicTypes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/CapturedTypeApproximation.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/constants/CompileTimeConstant.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/Utils/functions.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/AnnotationsImpl.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/full/KClasses.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/kotlin/methodSignatureBuilding.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2010-2015 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ScriptDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/header/ReadKotlinClassHeaderAnnotationVisitor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertySetterDescriptorImpl.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/JavaDescriptorUtil.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/KParameterImpl.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorNonRoot.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/StarProjectionImpl.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/utills/addToStdlib.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaCallableMemberDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationArgumentVisitor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassOrPackageFragmentDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/ReflectProperties.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/jvm/JvmClassName.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/SourceFile.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/Named.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/Visibility.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/LocalClassifierTypeSettings.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ImplicitClassReceiver.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/FieldOverridabilityCondition.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaModifierListOwner.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/ReflectionObjectRenderer.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/storage/ObservableStorageManager.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ParameterDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/MemberScope.kt

- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JvmPackagePartSource.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/SupertypeLoopChecker.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaClassDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/components/TypeUsage.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/RuntimeTypeMapper.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/ReflectionFactoryImpl.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/incremental/Utils.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/DeserializedClassDataFinder.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertyGetterDescriptorImpl.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotatedCallableKind.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DeclarationDescriptorImpl.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/JvmAnnotationNames.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/LazyPackageViewDescriptorImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotatedImpl.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ExtensionReceiver.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/jvm/JavaDescriptorResolver.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/LazyJavaAnnotations.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/MemberScopeImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/KPackageImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/overridingUtils.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ClassDeserializer.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/DescriptorFactory.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/checker/TypeCheckingProcedureCallbacks.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/Receiver.java

- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinJvmBinaryClass.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/FunctionWithAllInvokes.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/JavaIncompatibilityRulesOverridabilityCondition.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/JavaClassDataFinder.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyAccessorDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/DeserializedDescriptorResolver.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/CallableDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/DescriptorSubstitutor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ClassKind.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/pcollections/MapEntry.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinClassFinder.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ModuleDescriptorImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/FakePureImplementationsProvider.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ClassDataFinder.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/typeParameterUtils.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedPackageMemberScope.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ReceiverParameterDescriptorImpl.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/DisjointKeysUnionTypeSubstitution.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/builtins/CompanionObjectMapping.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/ExternalOverridabilityCondition.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationAndConstantLoader.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ConstUtil.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PackageFragmentDescriptorImpl.kt

- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/KCallablesJvm.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/SubpackagesScope.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageFragmentProviderImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ValueParameterDescriptorImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/reflectLambda.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/ChainedMemberScope.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/RuntimeErrorReporter.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/propertiesConventionUtil.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/descriptors/DeserializedMemberScope.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/EmptyContainerForLocal.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/VisibilityUtil.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptorWithVisibility.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/KClassImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/TypeProjection.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/components/SignaturePropagator.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/storage/NoLock.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/PropertyGetterDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/AnnotationAndConstantLoaderImpl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/AbstractReceiverValue.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaResolverCache.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/name/ClassId.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/MemberDescriptor.java
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/KProperty2Impl.kt
- * /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/ImplicitReceiver.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/utils/collections.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/kClassCache.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/CallableMemberDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/EffectiveVisibility.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/CompositePackageFragmentProvider.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/KotlinRetention.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/KClassesJvm.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/storage/StorageManager.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/DeclaredMemberIndex.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/name/SpecialNames.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/PackageFragmentDescriptor.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/TypeParameterDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ModuleDescriptor.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationUseSiteTarget.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/builtins/PrimitiveType.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/pcollections/ConsPStack.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/pcollections/HashPMap.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaPackageFragment.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/KDeclarationContainerImpl.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/DeclarationDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/AnnotationDescriptor.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/LazyClassReceiverParameterDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/JavaVisibilities.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/components/JavaAnnotationMapper.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/TypeConstructor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/kotlin/KotlinJvmBinarySourceElement.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/structure/annotationArguments.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/receivers/TransientReceiver.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaField.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2018 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/renderer/KeywordStringsGenerated.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2010-2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/Utils.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaClassifierType.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaStaticClassScope.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/ConstructorDescriptor.java

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/Utils/Jsr305State.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/load/java/components/SamConversionResolver.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/builtins/BuiltInsLoader.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/full/KProperties.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/checker/NewKotlinTypeChecker.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaWildcardType.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/types/checker/IntersectionType.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/container/DefaultImplementation.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaPackage.kt

* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-

jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/ClassConstructorDescriptorImpl.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaMethod.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeWithEnhancement.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaArrayType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/reflectClassUtil.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaConstructor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/DescriptorDerivedFromTypeAlias.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/ReflectKotlinClass.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/Modality.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/annotations/BuiltInAnnotationDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/SpecialTypes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/incremental/components/ExpectActualTracker.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/scopes/SyntheticScopes.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/descriptors/JavaClassConstructorDescriptor.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/AdditionalClassPartsProvider.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/util/modifierChecks.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/typeEnhancement/signatureEnhancement.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaValueParameter.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/ReflectKotlinClassFinder.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaMember.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/Substitutable.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaPrimitiveType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/ErrorType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-

jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/RuntimeModuleData.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaAnnotationArguments.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/KotlinType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/findClassInModule.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/ClassDescriptorFactory.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaClassMemberScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/AnnotationTypeQualifierResolver.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/TypeUtils.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaType.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/Visibilities.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/utills/numbers.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/lazy/descriptors/LazyJavaPackageScope.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/KotlinTypeFactory.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/DeserializedPackageFragmentImpl.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaAnnotationOwner.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaAnnotation.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/serialization/deserialization/ContractDeserializer.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/resolve/OverridingUtil.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/PropertyDescriptorImpl.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/ReflectJavaClassFinder.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/components/RuntimeSourceElementFactory.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/util/OperatorNameConventions.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/load/java/JavaClassesTracker.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/deserialization/PlatformDependentDeclarationFilter.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-

```
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaClass.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/config/KotlinCompilerVersion.java
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/runtime/structure/ReflectJavaTypeParameter.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/InvalidModuleException.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/descriptors/impl/TypeAliasConstructorDescriptor.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/builtins/BuiltInsPackageFragment.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/name/NameUtils.kt
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/types/checker/NewCapturedType.kt
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2010-2015 JetBrains s.r.o.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
// A seemingly obvious way to come about this case would be to declare a special exception class, but the problem is
that
```

```
Found in path(s):
* /opt/cola/permits/1136972820_1613985925.71/0/kotlin-reflect-1-3-71-sources-1-
jar/kotlin/reflect/jvm/internal/impl/storage/LockBasedStorageManager.java
```

1.9 pact-jvm-core-pact-broker 4.0.10

1.9.1 Available under license :

Apache-2.0

1.10 jansi 1.17.1

1.10.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2009-2017 the original author(s).
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiRenderer.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiMain.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiConsole.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiOutputStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiString.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/FilterPrintStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiPrintStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/HtmlAnsiOutputStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/WindowsAnsiOutputStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/Ansi.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/WindowsAnsiPrintStream.java
* /opt/ws_local/PERMITS_SQL/1082870243_1597967212.24/0/jansi-1-17-1-sources-1-
jar/org/fusesource/jansi/AnsiRenderWriter.java
```

1.11 okio 2.8.0

1.11.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-
jar/commonMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingTimeout.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2019 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-
```

jar/commonMain/okio/internal/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-
jar/commonMain/okio/internal/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-
jar/commonMain/okio/internal/SegmentedByteString.kt
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2017 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2019 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/Buffer.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Timeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/BufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Sink.kt

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Buffer.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/BufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Source.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Okio.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2014 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/SegmentPool.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/InflaterSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/BufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/JvmOkio.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Segment.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/BufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/GzipSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/DeflaterSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/GzipSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Buffer.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Source.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Sink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Timeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/SegmentPool.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ForwardingSink.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Util.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/ByteString.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Platform.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-Platform.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/internal/-Utf8.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/ByteString.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/ByteString.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUpgrade.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedUtf8.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/PeekSource.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/-DeprecatedOkio.kt
* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Throttler.kt
No license file was found, but licenses were detected in source scan.

/*
* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
*/

- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- * /

Found in path(s):

- * /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSink.kt
- * /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/Pipe.kt
- * /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/jvmMain/okio/HashingSource.kt
- * /opt/cola/permits/1274700344_1645235023.94/0/okio-2-8-0-sources-jar/commonMain/okio/Options.kt

1.12 hdrhistogram 2.1.9

1.12.1 Available under license :

Public Domain

1.13 micronaut 3.2.7

1.13.1 Available under license :

Micronaut includes references to parts of GraalVM 21.0 or later distributed under the UPL license below:

Copyright (c) 2018 - 2021 Oracle and/or its affiliates. All rights reserved

The Universal Permissive License (UPL), Version 1.0

Subject to the condition set forth below, permission is hereby granted to any person obtaining a copy of this software, associated documentation and/or data (collectively the "Software"), free of charge and under any and all copyright rights in the Software, and any and all patent rights owned or freely licensable by each licensor hereunder covering either (i) the unmodified Software as contributed to or provided by such licensor, or (ii) the Larger Works (as defined below), to deal in both

(a) the Software, and

(b) any piece of software and/or hardware listed in the lrgwrks.txt file if one is included with the Software (each a Larger Work to which the Software is contributed by such licensors),

without restriction, including without limitation the rights to copy, create derivative works of, display, perform, and distribute the Software and make, use, sell, offer for sale, import, export, have made, and have sold the Software and the Larger Work(s), and to sublicense the foregoing rights on either these or other terms.

This license is subject to the following condition:

The above copyright notice and either this complete permission notice or at a minimum a reference to the UPL must be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.14 micronaut-micrometer 4.2.0

1.14.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.15 pact-jvm-consumer-java8 4.0.10

1.15.1 Available under license :

Apache-2.0

1.16 appdynamics-java-agent-api 4.5.13.27526

1.16.1 Available under license :

No license file was found, but licenses were detected in source scan.

<div class="aboutLanguage">Copyright 2019 AppDynamics Inc. All rights reserved.</div>

Found in path(s):

* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/overview-tree.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/package-tree.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/ExitCall.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/deprecated-list.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/impl/NoOpTransaction.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/Transaction.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/instrumentation/sdk/logging/package-tree.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/instrumentation/sdk/logging/package-summary.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/AppdynamicsAgent.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/instrumentation/sdk/logging/ISDKLogger.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/EventPublisher.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/impl/package-summary.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/EntryTypes.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/overview-summary.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/constant-values.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/package-summary.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/index-all.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/ExitTypes.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/help-doc.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/impl/NoOpExitCall.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/impl/package-tree.html
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-javadoc-jar/com/appdynamics/agent/api/MetricPublisher.html

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) AppDynamics, Inc., and its affiliates

* 2018
* All Rights Reserved
* THIS IS UNPUBLISHED PROPRIETARY CODE OF APPDYNAMICS, INC.
* The copyright notice above does not evidence any actual or intended publication of such source code
*/

Found in path(s):

* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/impl/NoOpTransaction.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/EumDelegate.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/impl/NoOpExitCall.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/AppdynamicsAgent.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/ExitCall.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/EventPublisher.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/MetricPublisher.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/Transaction.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) AppDynamics, Inc., and its affiliates
* 2019
* All Rights Reserved
* THIS IS UNPUBLISHED PROPRIETARY CODE OF APPDYNAMICS, INC.
* The copyright notice above does not evidence any actual or intended publication of such source code
*/

Found in path(s):

* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/bootstrap/NoOpTransactionDelegate.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/ExitTypes.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/EntryTypes.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/bootstrap/IApiBootstrapFacade.java
* /opt/cola/permits/1337070300_1654792344.40951/0/java-agent-api-4-5-13-27526-zip/agent-api-sources-jar/com/appdynamics/agent/api/bootstrap/IApiTransactionDelegate.java

1.17 kafka-schema-serializer 5.5.1

1.17.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://maven.apache.org/POM/4.0.0"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<parent>
  <groupId>io.confluent</groupId>
  <artifactId>kafka-schema-registry-parent</artifactId>
  <version>5.5.1</version>
</parent>
```

```
<licenses>
  <license>
    <name>Apache License 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
    <distribution>repo</distribution>
  </license>
</licenses>
```

```
<artifactId>kafka-schema-serializer</artifactId>
<packaging>jar</packaging>
<name>kafka-schema-serializer</name>
```

```
<dependencies>
  <dependency>
    <groupId>org.apache.kafka</groupId>
    <artifactId>kafka_${kafka.scala.version}</artifactId>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-client</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>common-config</artifactId>
  </dependency>
```

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
```

```
<scope>test</scope>
</dependency>
</dependencies>
</project>
```

Found in path(s):

```
* /opt/cola/permits/1337069416_1654119464.16387/0/kafka-schema-serializer-5-5-1-jar/META-INF/maven/io.confluent/kafka-schema-serializer/pom.xml
```

1.18 io-swagger-swagger-annotations 1.6.2

1.18.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
Manifest-Version: 1.0
Bnd-LastModified: 1593602041104
Build-Jdk: 1.8.0_151
Built-By: frantuma
Bundle-Description: Sonatype helps open source projects to set up Maven repositories on https://oss.sonatype.org/
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.html
Bundle-ManifestVersion: 2
Bundle-Name: swagger-annotations
Bundle-SymbolicName: io.swagger.annotations
Bundle-Version: 1.6.2
Created-By: Apache Maven Bundle Plugin
Export-Package: io.swagger.annotations;version="1.6.2"
implementation-version: 1.6.2
mode: development
package: io.swagger
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.3.0.201609221906
url: https://github.com/swagger-api/swagger-core/modules/swagger-annotations
```

Found in path(s):

```
* /opt/cola/permits/1170799104_1654737045.2729578/0/swagger-annotations-1-6-2-jar/META-INF/MANIFEST.MF
```

1.19 elasticsearch-kubernetes-cloud-plugin

4.10.3

1.19.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.20 hk2-api-module 2.6.1

1.20.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement

or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights

in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software

or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE

file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute

such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1

and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new

versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it

free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: <https://projects.eclipse.org/projects/ee4j.glassfish>

Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at

<http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/glassfish-ha-api>
- * <https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor>
- * <https://github.com/eclipse-ee4j/glassfish-shoal>
- * <https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck>
- * <https://github.com/eclipse-ee4j/glassfish-jsftemplating>
- * <https://github.com/eclipse-ee4j/glassfish-hk2-extra>
- * <https://github.com/eclipse-ee4j/glassfish-hk2>
- * <https://github.com/eclipse-ee4j/glassfish-fighterfish>

Third-party Content

This project leverages the following third party content.

None

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.21 apache-kafka 3.0.0

1.21.1 Available under license :

Apache Kafka

Copyright 2021 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at <https://github.com/jersey/jersey/>.

This distribution has a binary test dependency on jqwik, which is available under the Eclipse Public License 2.0. The source code can be found at <https://github.com/jlink/jqwik>.

The streams-scala (streams/streams-scala) module was donated by Lightbend and the original code was copyrighted by them:

Copyright (C) 2018 Lightbend Inc. <<https://www.lightbend.com>>

Copyright (C) 2017-2018 Alexis Seigneurin.

This project contains the following code copied from Apache Hadoop:

clients/src/main/java/org/apache/kafka/common/utils/PureJavaCrc32C.java

Some portions of this file Copyright (c) 2004-2006 Intel Corporation and licensed under the BSD license.

This project contains the following code copied from Apache Hive:

streams/src/main/java/org/apache/kafka/streams/state/internals/Murmur3.java

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.22 annotations-for-metrics 4.1.17

1.22.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.codahale.metrics

Build-Jdk-Spec: 1.8

Bundle-Description: Metrics is a Java library which gives you unparalleled insight into what your code does in production. Metrics provides a powerful toolkit of ways to measure the behavior of critical components in your production environment.

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>

Bundle-ManifestVersion: 2

Bundle-Name: Metrics Core

Bundle-SymbolicName: io.dropwizard.metrics.core

Bundle-Version: 4.1.17

Created-By: Apache Maven Bundle Plugin

Export-Package: com.codahale.metrics;uses:="org.slf4j";version="4.1.17"

"

Implementation-Title: Metrics Core
Implementation-Version: 4.1.17
Import-Package: org.slf4j;version="[1.6.0,2.0.0)"
Private-Package: com.codahale.metrics
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188463_1654750948.3729246/0/metrics-core-4-1-17-jar/META-INF/MANIFEST.MF

1.23 fabric8-::-kubernetes-model-::-storage-class 4.10.3

1.23.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
# Licensed under the Apache License, Version 2.0 (the "License");  
# you may not use this file except in compliance with the License.  
# You may obtain a copy of the License at  
#   http://www.apache.org/licenses/LICENSE-2.0  
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

* /opt/cola/permits/1337069668_1654218635.2626/0/kubernetes-model-storageclass-4-10-3-jar/ValidationMessages.properties

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Bnd-LastModified: 1594746520780
Build-Jdk: 14.0.1
Build-Timestamp: \${build.datetime}
Built-By: rohaan
Bundle-Description: Java client for Kubernetes and OpenShift
Bundle-DocURL: http://redhat.com
Bundle-License: http://www.apache.org/licenses/LICENSE-2.0.txt
Bundle-ManifestVersion: 2
Bundle-Name: Fabric8 :: Kubernetes Model :: Storage Class
Bundle-SymbolicName: io.fabric8.kubernetes-model-storageclass
Bundle-Vendor: Red Hat
Bundle-Version: 4.10.3
Created-By: Apache Maven Bundle Plugin
Export-Package: io.fabric8.kubernetes.api.model.storage;uses:="com.fasterxml.jackson.annotation,com.fasterxml.jackson.databind,com.fasterxml.jackson.databind.annotation,io.fabric8.kubernetes.api.builder,io.fabric8.kubernetes.api.model,io.fabric8.kubernetes.model.annotation";version="4.10.3",io.fabric8.kubernetes.api.model.storage.v1beta1;uses:="

"com.fasterxml.jackson.annotation,com.fasterxml.jackson.databind,com.fasterxml.jackson.databind.annotation,io.fabric8.kubernetes.api.builder,io.fabric8.kubernetes.api.model,io.fabric8.kubernetes.model.annotation";version="4.10.3"

Implementation-Title: Fabric8 :: Kubernetes Model :: Storage Class

Implementation-URL: <http://fabric8.io/kubernetes-model-generator/kubernetes-model-storageclass/>

Implementation-Vendor: Red Hat

Implementation-Vendor-Id: io.fabric8

Implementation-Version: 4.10.3

Import-Package: com.fasterxml.jackson.annotation;version="[2.10,3)",com.fasterxml.jackson.databind;version="[2.10,3)",com.fasterxml.jackson.databind.annotation;version="[2.10,3)",io.fabric8.kubernetes.api.builder;version="[4.10,5)",io.fabric8.kubernetes.api.model,io.fabric8.kubernetes.model.annotation;version="[4.10,5)"

Java-Vendor: Oracle Corporation

Java-Version: 14.0.1

Os-Arch: amd64

Os-Name: Linux

Os-Version: 5.6.19-300.fc32.x86_64

Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"

Scm-Connection: scm:git:git@github.com:fabric8io/kubernetes-client.git/kubernetes-model-generator/kubernetes-model-storageclass

Scm-Revision: \${buildNumber}

Scm-Url: <http://github.com/fabric8io/kubernetes-client/kubernetes-model-generator/kubernetes-model-storageclass/>

Specification-Title: Fabric8 :: Kubernetes Model :: Storage Class

Specification-Vendor: Red Hat

Specification-Version: 4.10.3

Tool: Bnd-4.2.0.201903051501

Found in path(s):

* /opt/cola/permits/1337069668_1654218635.2626/0/kubernetes-model-storageclass-4-10-3-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

*

* Copyright (C) 2015 Red Hat, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and
* limitations under the License.
*

Found in path(s):

* /opt/cola/permits/1337069668_1654218635.2626/0/kubernetes-model-storageclass-4-10-3-jar/manifest.vm

No license file was found, but licenses were detected in source scan.

<!--

Copyright (C) 2015 Red Hat, Inc.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1337069668_1654218635.2626/0/kubernetes-model-storageclass-4-10-3-jar/META-

INF/maven/io.fabric8/kubernetes-model-storageclass/pom.xml

1.24 reactor-core 3.4.14

1.24.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2016-2021 VMware Inc. or its affiliates, All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSwitchIfEmpty.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxBufferBoundary.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxHandleFuseable.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSingle.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxConcatIterable.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxFlatMap.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/DirectProcessor.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSupplier.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxZipIterable.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoReduceSeed.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTakeLastOne.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxContextWrite.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelMergeSequential.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelMergeReduce.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple8.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelReduceSeed.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDefaultIfEmpty.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InternalFluxOperator.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoEmpty.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxCombineLatest.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxOnErrorResume.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/publisher/MonoDoFinally.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxTake.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelPeek.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/SerializedSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxErrorOnRequest.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoHasElements.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSwitchMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxTakeUntilOther.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSubscribeOnCallable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoCollect.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/InternalMonoOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoFromPublisher.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxErrorSupplied.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/Operators.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSkip.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoMapFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/BaseSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelSource.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxPublish.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/Schedulers.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDematerialize.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxMapSignal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/NonBlocking.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/scheduler/ParallelScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxHandle.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/DelegateProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoOnErrorResume.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDelay.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDoFinally.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoFilterFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSkipLast.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/SignalType.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDoOnEach.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoCount.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDoFirst.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxConcatArray.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDefer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxFlattenIterable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/InnerProducer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/SinkOneMulticast.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/InnerConsumer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/Hooks.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxMerge.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoZip.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoLiftFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxPeekFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/publisher/MonoCompletionStage.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxOnBackpressureLatest.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoLift.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxRetryWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSingleMono.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxFromMonoOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxInterval.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSource.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/Mono.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSubscribeOnCallable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxScan.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSink.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoAll.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoFilter.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelLiftFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/Logger.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/ImmediateScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoFlatMapMany.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSourceFluxFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSwitchIfEmpty.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxEmpty.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSubscribeOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelDoOnEach.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoTimed.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDoOnEachFuseable.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSourceFlux.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/Flux.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRepeatPredicate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoContextWrite.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/SingleWorkerScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSourceMono.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxMapFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SignalPeek.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/EmitterProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelCollect.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWindowPredicate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuples.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCollectList.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDistinct.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoPeekFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxPublishOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxLogFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSubscribeOnValue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SourceProducer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/Fuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxCallableOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoPeek.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/UnicastProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/NextProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelFluxOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDoFirstFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxGroupBy.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple2.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxUsing.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple5.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCallable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSourceFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ReplayProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRetry.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/Loggers.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTakeFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxError.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxCreate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoLogFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoSingleCallable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxFirstWithSignal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWithLatestFrom.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoElementAt.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxFirstWithValue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/DrainUtils.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxGroupJoin.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/publisher/FluxNameFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelFluxName.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxName.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDelaySubscription.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDoFirst.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxLiftFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxRepeat.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSkipWhile.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDefer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoNameFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDoFinallyFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxStream.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxReplay.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSourceFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoBridges.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelFlux.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSequenceEqual.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/util/function/Tuple4.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/ExecutorScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDelaySubscription.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/util/concurrent/SpscLinkedArrayQueue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoAny.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ConnectableLiftFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDetach.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/publisher/FluxFilter.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoError.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoHasElement.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxBufferWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/BlockingSingleSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoDeferContextual.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxElapsed.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSample.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxRetry.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxDoFinallyFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelFilter.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoFromFluxOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoToCompletableFuture.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSubscribeOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoMaterialize.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/LambdaMonoSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelMergeSort.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoElapsed.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ConnectableLift.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxAutoConnectFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoCurrentContext.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/Metrics.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoLog.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoHandleFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxMap.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/GroupedLiftFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDistinctFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCancelOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BlockingOptionalMonoSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTakeUntil.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSkipUntil.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxBufferTimeout.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/SingleScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/Scheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDematerialize.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRetryWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoTimeout.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoNever.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxScanSeed.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoName.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoTakeUntilOther.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxRepeatPredicate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRepeatWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxOnBackpressureDrop.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDistinctUntilChanged.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxRange.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelLift.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoIgnoreElements.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSubscribeOnValue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoDefaultIfEmpty.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxRepeatWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/OptimizableOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxAutoConnect.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTakeWhile.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoHandle.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SignalLogger.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/CorePublisher.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxConcatMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSourceMonoFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoPeekTerminal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelRunOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxMergeSequential.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFirstWithValue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSource.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ConnectableFluxHide.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoNext.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple6.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRepeat.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxCallable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple3.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDetach.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxGenerate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxBufferPredicate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelConcatMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoIgnoreThen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxPeek.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelLog.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxRefCount.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/Exceptions.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoDoOnEach.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/concurrent/SpSCArrayQueue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoErrorSupplied.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/Signal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoDelayElement.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ImmutableSignal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/LambdaSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoHide.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoStreamCollector.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxIndex.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFlattenIterable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxMaterialize.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BlockingFirstSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/ElasticScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSkipUntilOther.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDoOnEachFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxIndexFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxPublishMulticast.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelGroup.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BlockingIterable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxLift.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxFilterFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoPublishOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoDoFirstFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxIterable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelFlatMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxZip.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoProcessor.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/adapters/JdkFlowAdapter.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ConnectableFlux.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxBuffer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxHide.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFirstWithSignal.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelFluxHide.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxLog.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/DirectInnerContainer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BlockingMonoSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoTakeLastOne.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDeferContextual.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWindow.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxArray.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SynchronousSink.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCreate.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelMergeOrdered.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWindowWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoUsing.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ConnectableFluxOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFlatMap.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/Tuple7.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCallableOnAssembly.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoReduce.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InternalConnectableFluxOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxNever.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSampleFirst.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxSampleTimeout.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTimeout.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoIgnoreElement.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxJoin.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxCancelOn.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoIgnorePublisher.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWindowBoundary.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTakeLast.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/GroupedLift.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (c) 2021 VMware Inc. or its affiliates, All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoCacheInvalidateIf.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoCacheInvalidateWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSwitchMapNoPrefetch.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (c) 2015-2021 VMware Inc. or its affiliates, All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxOnBackpressureBuffer.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoSink.java
```

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/ExecutorServiceWorker.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/GroupedFlux.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoJust.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context3.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context2.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BlockingLastSubscriber.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InnerOperator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxOnBackpressureBufferStrategy.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/ContextN.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/BufferOverflowStrategy.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context0.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoRunnable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ContextHolder.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/Disposable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxJust.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context1.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/ParallelArraySource.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2002-2017 the original author or authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/annotation/Nullable.java
 - * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/annotation/NonNullApi.java
 - * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/annotation/NonNull.java
- No license file was found, but licenses were detected in source scan.

/*

- * Copyright (c) 2018-2022 VMware Inc. or its affiliates, All Rights Reserved.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <https://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoUsingWhen.java
 - * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxUsingWhen.java
- No license file was found, but licenses were detected in source scan.

/*

- * Copyright (c) 2020-2021 VMware Inc. or its affiliates, All Rights Reserved.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <https://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinkEmptySerialized.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InternalManySink.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/Timed.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/UnicastManySinkNoBackpressure.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/retry/Retry.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinkManySerialized.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxConcatMapNoPrefetch.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinkEmptyMulticast.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinkManyBestEffort.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/retry/RetrySpec.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/ContextView.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/Sinks.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/DelegatingScheduledExecutorService.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/retry/RetryBackoffSpec.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinkOneSerialized.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InternalEmptySink.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxTimed.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/InternalOneSink.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/SinksSpecs.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/retry/ImmutableRetrySignal.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2011-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/adapters/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoExtensions.kt
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/TupleExtensions.kt
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/concurrent/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxExtensions.kt
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFunctions.kt
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/function/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/package-info.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/package-info.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (c) 2019-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

```
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ParallelThen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/ContextTrackingFunctionWrapper.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/BoundedElasticScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/util/context/CoreContext.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/ReactorBlockHoundIntegration.java
No license file was found, but licenses were detected in source scan.
```

```
# Copyright (c) 2019-2021 VMware Inc. or its affiliates, All Rights Reserved.
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
reactor.core.scheduler.ReactorBlockHoundIntegration
```

Found in path(s):

```
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/META-
INF/services/reactor.blockhound.integration.BlockHoundIntegration
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright (c) 2017-2021 VMware Inc. or its affiliates, All Rights Reserved.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
```

- * You may obtain a copy of the License at
- *
- * <https://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/Disposables.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/InstantPeriodicWorkerTask.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxRefCountGrace.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/EmptyCompositeDisposable.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/CoreSubscriber.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoWhen.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/PeriodicSchedulerTask.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context5.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxExpand.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxDelaySequence.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/WorkerTask.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/StrictSubscriber.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoPublishMulticast.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxOnBackpressureBufferTimeout.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/Scannable.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoDelayUntil.java
- * /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/context/Context4.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/util/concurrent/Queues.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoCacheTime.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoFilterWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxWindowTimeout.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/SchedulerTask.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxFilterWhen.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/OperatorDisposables.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/PeriodicWorkerTask.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/scheduler/DelegateServiceScheduler.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxLimitRequest.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/MonoExpand.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2018-2021 VMware Inc. or its affiliates, All Rights Reserved.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxMetrics.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/QueueDrainSubscriber.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-jar/reactor/core/publisher/FluxMetricsFuseable.java

* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-

jar/reactor/core/publisher/FluxMergeComparing.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoMetricsFuseable.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/SchedulerMetricDecorator.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/MonoMetrics.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/FluxSwitchOnFirst.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/OnNextFailureStrategy.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/util/concurrent/MpscLinkedQueue.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/publisher/Traces.java
* /opt/cola/permits/1340076475_1654622420.5909758/0/reactor-core-3-4-14-sources-1-
jar/reactor/core/scheduler/ReactorThreadFactory.java

1.25 jakarta-ws-rs-api 3.0.0

1.25.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which

are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall

apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness

for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged

intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of

the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s)},

version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis

or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other

recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not

signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notices for Jakarta RESTful Web Services

This content is produced and maintained by the ****Jakarta RESTful Web Services**** project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jaxrs>

Trademarks

****Jakarta RESTful Web Services**** is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jaxrs-api>

Third-party Content

This project leverages the following third party content.

javaee-api (7.0)

* License: Apache-2.0 AND W3C

JUnit (4.11)

* License: Common Public License 1.0

Mockito (2.16.0)

* Project: <http://site.mockito.org>

* Source: <https://github.com/mockito/mockito/releases/tag/v2.16.0>

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.26 wire-schema 3.2.2

1.26.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2019 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/jvmMain/com/squareup/wire/schema/CoreLoader.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/commonMain/com/squareup/wire/schema/FileLinker.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/commonMain/com/squareup/wire/schema/Loader.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/commonMain/com/squareup/wire/schema/SemVer.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/jvmMain/com/squareup/wire/schema/internal/UtilJVM.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2016 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```


- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/EnclosingType.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ReservedElement.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/GroupElement.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Reserved.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/SyntaxReader.kt

No license file was found, but licenses were detected in source scan.

/*

- * Copyright (C) 2015 Square, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/MessageType.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/SchemaException.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Extensions.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Service.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/ProtoFile.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/jvmMain/com/squareup/wire/schema/SchemaLoader.kt
- * /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/OneOf.kt

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Rpc.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Schema.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/EnumType.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/SchemaProtoAdapterFactory.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Extend.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Linker.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Type.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/MarkSet.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Field.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Location.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/EnumConstant.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/PruningRules.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/ProtoMember.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Pruner.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Options.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/ProtoType.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2014 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/OneOfElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/EnumConstantElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/FieldElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/RpcElement.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ProtoParser.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/EnumElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ExtensionsElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ExtendElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/Util.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/TypeElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ProtoFileElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/OptionElement.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/MessageElement.kt

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/ServiceElement.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/internal/parser/OptionReader.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/CoreLoader.kt

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/Multimap.kt

* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-jar/commonMain/com/squareup/wire/schema/SyntaxRules.kt

```
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/jvmMain/com/squareup/wire/schema/ClaimedDefinitions.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/jvmMain/com/squareup/wire/schema/EmittingRules.kt
* /opt/cola/permits/1337069608_1654119466.6796012/0/wire-schema-3-2-2-sources-
jar/jvmMain/com/squareup/wire/schema/Multimap.kt
```

1.27 wire-protocol-buffer-runtime 3.6.0

1.27.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2016 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/jvmMain/com/squareup/wire/EnumAdapter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/jvmMain/com/squareup/wire/AndroidMessage.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/commonMain/com/squareup/wire/EnumAdapter.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
```

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/internal/Util.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2020 Square Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/OneOf.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/Instant.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/Duration.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/EnumJsonFormatter.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/WireEnumConstant.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/Duration.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/Instant.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/DurationJsonFormatter.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/JsonFormatter.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/Syntax.kt

```
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/jvmMain/com/squareup/wire/internal/InstantJsonFormatter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/commonMain/com/squareup/wire/AnyMessage.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/jvmMain/com/squareup/wire/internal/JsonIntegration.kt
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright (c) 2016, the R8 project authors.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions are met:
*
* * Redistributions of source code must retain the above copyright notice, this
* list of conditions and the following disclaimer.
*
* * Redistributions in binary form must reproduce the above copyright notice,
* this list of conditions and the following disclaimer in the documentation
* and/or other materials provided with the distribution.
*
* * Neither the name of Google Inc. nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
* SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER
* CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
* OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

Found in path(s):

```
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/commonMain/com/squareup/wire/internal/MathMethods.kt
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2013 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
```

* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

// Copyright 2013 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/commonMain/com/squareup/wire/ProtoWriter.kt

No license file was found, but licenses were detected in source scan.

// Copyright 2013 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-
jar/commonMain/com/squareup/wire/ProtoReader.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 Square Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/internal/MutableOnWriteList.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/FieldBinding.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/ProtoAdapter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/ProtoAdapter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/MessageSerializedForm.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2019 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/internal/Internal.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/MessageSink.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/-Platform.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/WireRpc.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/Service.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/MessageSink.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/internal/-Platform.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/MessageSource.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/MessageSource.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2013 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/RuntimeEnumAdapter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/RuntimeMessageAdapter.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/FieldEncoding.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/Message.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/internal/ImmutableList.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/InternalJvm.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/WireEnum.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/Message.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/commonMain/com/squareup/wire/WireField.kt
* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-

jar/jvmMain/com/squareup/wire/Wire.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2021 Square Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/OneOfBinding.kt

* /opt/cola/permits/1340031473_1654666144.907512/0/wire-runtime-3-6-0-sources-jar/jvmMain/com/squareup/wire/internal/FieldOrOneOfBinding.kt

1.28 animal-sniffer-annotation 1.19

1.28.1 Available under license :

No license file was found, but licenses were detected in source scan.

The MIT License

Copyright (c) 2009 codehaus.org.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Found in path(s):

* /opt/cola/permits/1127519953_1611825926.02/0/animal-sniffer-annotations-1-19-sources-jar/META-INF/maven/org.codehaus.mojo/animal-sniffer-annotations/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* The MIT License

*

* Copyright (c) 2008 Kohsuke Kawaguchi and codehaus.org.

*

* Permission is hereby granted, free of charge, to any person obtaining a copy

* of this software and associated documentation files (the "Software"), to deal

* in the Software without restriction, including without limitation the rights

* to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

* copies of the Software, and to permit persons to whom the Software is

* furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be included in

* all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

* AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER

* LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

* OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN

* THE SOFTWARE.

*

*/

Found in path(s):

* /opt/cola/permits/1127519953_1611825926.02/0/animal-sniffer-annotations-1-19-sources-jar/org/codehaus/mojo/animal_sniffer/IgnoreJRERequirement.java

1.29 checker-qual 3.12.0

1.29.1 Available under license :

Checker Framework qualifiers

Copyright 2004-present by the Checker Framework developers

MIT License:

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.30 generex 1.0.2

1.30.1 Available under license :

No license file was found, but licenses were detected in source scan.

Generex is licensed under the Apache License, Version 2.0.

<http://www.apache.org/licenses/LICENSE-2.0>

Found in path(s):

```
* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/README.md
```

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.github.mifmif</groupId>
  <artifactId>generex</artifactId>
  <version>1.0.2</version>
  <name>Generex</name>
  <url>https://github.com/mifmif/Generex/tree/master</url>
  <description>Generex A Java Library for regex to Strings generation</description>
  <packaging>jar</packaging>
  <licenses>
    <license>
      <name>The Apache Software License, Version 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
      <distribution>repo</distribution>
    </license>
  </licenses>
```

```

</licenses>
<developers>
  <developer>
    <id>mifmif</id>
    <name>MIFRAH Youssef</name>
    <email>mifmif.com@gmail.com</email>
  </developer>
  <developer>
    <id>mkolisnyk</id>
    <name>mkolisnyk</name>
    <email>kolesnik.nickolay@gmail.com</email>
  </developer>
</developers>

<parent>
  <groupId>org.sonatype.oss</groupId>
  <artifactId>oss-parent</artifactId>
  <version>7</version>
</parent>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.pmd.excludes>*/generated-sources/*</maven.pmd.excludes>
</properties>

<scm>
  <connection>scm:git:git@github.com:mifmif/Generex.git</connection>
  <developerConnection>scm:git:git@github.com:mifmif/Generex.git</developerConnection>
  <url>https://github.com/mifmif/Generex.git</url>
</scm>

<ciManagement>
  <url>https://travis-ci.org/mifmif/Generex/builds</url>
  <system>Travis</system>
  <notifiers>
    <notifier>
      <address>mifmif.com@gmail.com</address>
      <type>mail</type>
    </notifier>
    <notifier>
      <address>kolesnik.nickolay@gmail.com</address>
      <type>mail</type>
    </notifier>
  </notifiers>
</ciManagement>

<distributionManagement>
  <snapshotRepository>
    <id>ossrh</id>
    <url>https://oss.sonatype.org/content/repositories/snapshots</url>
  </snapshotRepository>
</distributionManagement>

```

```
</snapshotRepository>
<repository>
  <id>ossrh</id>
  <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
</repository>
</distributionManagement>
```

```
<build>
  <sourceDirectory>src/main/java</sourceDirectory>
  <testSourceDirectory>src/test/java</testSourceDirectory>
  <resources>
    <resource>
      <directory>src</directory>
      <excludes>
        <exclude>**/*.java</exclude>
      </excludes>
    </resource>
    <resource>
      <directory>target/dependency</directory>
      <excludes>
        <exclude>**/*.java</exclude>
      </excludes>
    </resource>
    <resource>
      <directory>src/main/resources</directory>
      <includes>
        <include>**/*.properties</include>
      </includes>
    </resource>
  </resources>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-install-plugin</artifactId>
      <version>2.5.2</version>
      <configuration>
        <file>target/${project.artifactId}-${project.version}.jar</file>
        <groupId>${project.groupId}</groupId>
        <artifactId>${project.artifactId}</artifactId>
        <version>${project.version}</version>
        <packaging>maven-plugin</packaging>
      </configuration>
    </plugin>

    <plugin>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.5.1</version>
```

```

<configuration>
  <source>1.5</source>
  <target>1.5</target>
</configuration>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
  <configuration>
    <excludes>
      <exclude>*</exclude>
      <exclude>com/thoughtworks/**/*</exclude>
      <exclude>freemarker/**/*</exclude>
      <exclude>ftl/**/*</exclude>
      <exclude>i18n/**/*</exclude>
      <exclude>style/**/*</exclude>
      <exclude>junit/**/*</exclude>
      <exclude>licenses/**/*</exclude>
      <exclude>META-INF/maven/**/*</exclude>
      <exclude>org/codehaus/**/*</exclude>
      <exclude>org/hamcrest/**/*</exclude>
      <exclude>org/jbehave/**/*</exclude>
      <exclude>org/junit/**/*</exclude>
      <exclude>org/testng/**/*</exclude>
      <exclude>org/xmlpull/**/*</exclude>
      <exclude>stories/**/*</exclude>
      <exclude>style/**/*</exclude>
      <exclude>tests/**/*</exclude>
    </excludes>
    <archive>
      <manifest>
        <addClasspath>>false</addClasspath>
        <addDefaultImplementationEntries>>true</addDefaultImplementationEntries>
        <addDefaultSpecificationEntries>>true</addDefaultSpecificationEntries>
        <addExtensions>>false</addExtensions>
        <classpathLayoutType>simple</classpathLayoutType>
      </manifest>
    </archive>
  </configuration>
</plugin>
<plugin>
  <artifactId>maven-assembly-plugin</artifactId>
  <version>2.6</version>
  <configuration>
    <descriptors>
      <descriptor>./sources.xml</descriptor>
    </descriptors>
  </configuration>

```



```

</configuration>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-dependency-plugin</artifactId>
  <version>2.10</version>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-javadoc-plugin</artifactId>
  <version>2.10.3</version>
  <configuration>
    <show>private</show>
    <nohelp>>false</nohelp>
    <failOnError>>false</failOnError>
    <sourcepath>src/main/java</sourcepath>
  </configuration>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-release-plugin</artifactId>
  <configuration>
    <tagNameFormat>v@{project.version}</tagNameFormat>
    <preparationGoals>package install:install-file</preparationGoals>
    <scmCommentPrefix>#3</scmCommentPrefix>
  </configuration>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-deploy-plugin</artifactId>
  <version>2.8.2</version>
  <configuration>
    <skip>>true</skip>
  </configuration>
</plugin>

<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-source-plugin</artifactId>
  <version>3.0.1</version>
  <executions>
    <execution>
      <goals>
        <goal>jar-no-fork</goal>
      </goals>
    </execution>
  </executions>

```

</plugin>

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-javadoc-plugin</artifactId>
  <executions>
    <execution>
      <id>attach-javadocs</id>
      <goals>
        <goal>jar</goal>
      </goals>
    </execution>
  </executions>
</plugin>
```

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-gpg-plugin</artifactId>
  <executions>
    <execution>
      <id>sign-artifacts</id>
      <phase>deploy</phase>
      <goals>
        <goal>sign</goal>
      </goals>
    </execution>
  </executions>
```

</plugin>

```
<plugin>
  <groupId>org.sonatype.plugins</groupId>
  <artifactId>nexus-staging-maven-plugin</artifactId>
  <version>1.6.3</version>
  <executions>
    <execution>
      <id>default-deploy</id>
      <phase>deploy</phase>
      <goals>
        <goal>deploy</goal>
      </goals>
    </execution>
  </executions>
  <extensions>true</extensions>
  <configuration>
```

```

<serverId>ossrh</serverId>
<nexusUrl>https://oss.sonatype.org/</nexusUrl>
</configuration>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-pmd-plugin</artifactId>
<version>3.6</version>
<configuration>
<failOnViolation>>true</failOnViolation>
<excludeRoots>
<excludeRoot>target/generated-sources/plugin</excludeRoot>
</excludeRoots>
<excludes>
<exclude>**/HelpMojo.java</exclude>
</excludes>
</configuration>
<executions>
<execution>
<goals>
<goal>check</goal>
<goal>cpd-check</goal>
</goals>
<configuration>
<failOnViolation>>true</failOnViolation>
<excludes>
<exclude>**/HelpMojo.java</exclude>
</excludes>
</configuration>
</execution>
</executions>
</plugin>
<plugin>
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-checkstyle-plugin</artifactId>
<version>2.10</version>
<configuration>
<excludes>**/test/java/**/*.*.java</excludes>
<configLocation>./sun_checks.xml</configLocation>
<suppressionsLocation>./checkstyle-suppressions.xml</suppressionsLocation>
<failOnViolation>>true</failOnViolation>
</configuration>
<executions>
<execution>
<goals>
<goal>check</goal>
</goals>
</execution>

```

```

</executions>
</plugin>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>cobertura-maven-plugin</artifactId>
<version>2.6</version>
<configuration>
<check>
<haltOnFailure>>false</haltOnFailure>
<totalBranchRate>80</totalBranchRate>
<totalLineRate>80</totalLineRate>
<packageLineRate>80</packageLineRate>
<packageBranchRate>80</packageBranchRate>
</check>
<instrumentation>
<excludes>
<exclude>**/HelpMojo.class</exclude>
</excludes>
</instrumentation>
</configuration>
<executions>
<execution>
<id>verification</id>
<phase>verify</phase>
<goals>
<goal>check</goal>
<goal>cobertura</goal>
</goals>
</execution>
</executions>
</plugin>
<plugin>
<groupId>org.codehaus.mojo</groupId>
<artifactId>javancss-maven-plugin</artifactId>
<version>2.0</version>
<configuration>
<encoding>${project.build.sourceEncoding}</encoding>
<ccnLimit>10</ccnLimit>
<ncssLimit>100</ncssLimit>
<failOnViolation>>true</failOnViolation>
<excludes>
<exclude>**/tests/**/*.*</exclude>
</excludes>
</configuration>
<executions>
<execution>
<goals>
<goal>check</goal>

```

```

    </goals>
  </execution>
</executions>
</plugin>
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>findbugs-maven-plugin</artifactId>
  <version>3.0.3</version>
  <configuration>
    <encoding>${project.build.sourceEncoding}</encoding>
    <failOnError>true</failOnError>
    <excludeFilterFile>findBugsExclude.xml</excludeFilterFile>
  </configuration>
  <executions>
    <execution>
      <goals>
        <goal>check</goal>
      </goals>
    </execution>
  </executions>
</plugin>
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>jdepend-maven-plugin</artifactId>
  <version>2.0</version>
</plugin>
</plugins>
</build>
<dependencies>
  <dependency>
    <groupId>dk.brics.automaton</groupId>
    <artifactId>automaton</artifactId>
    <version>1.11-8</version>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.hamcrest</groupId>
    <artifactId>hamcrest-junit</artifactId>
    <version>2.0.0.0</version>
    <scope>test</scope>
  </dependency>
</dependencies>
</project>

```

Found in path(s):

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 y.mifrah

*

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterable.java

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/GenerexIterator.java

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/util/Iterator.java

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Node.java

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Main.java

* /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/main/java/com/mifmif/common/regex/Generex.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 y.mifrah

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.

*/

Found in path(s):

- * /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexUnitTest.java

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2015 y.mifrah

*

- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at

*

- * <http://www.apache.org/licenses/LICENSE-2.0>

*

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.

*/

Found in path(s):

- * /opt/cola/permits/1162863542_1620923914.63/0/mifmif-generex-1-0-2-0-ge910243-tar-gz/mifmif-Generex-e910243/src/test/java/com/mifmif/common/regex/GenerexIteratorUnitTest.java

1.31 feign-jaxrs 8.18.0

1.31.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2013 Netflix, Inc.

*

- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at

*

- * <http://www.apache.org/licenses/LICENSE-2.0>

*

- * Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

*/opt/cola/permits/1274698784_1645233710.69/0/feign-jaxrs-8-18-0-sources-jar/feign/jaxrs/JAXRSContract.java

1.32 antlr-4-tool 4.7.2

1.32.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
[The "BSD licence"]  
Copyright (c) 2006 Kay Roepke  
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/  
/*
```

This file contains the actual layout of the messages emitted by ANTLR.
The text itself is coming out of the languages/*stg files, according to the chosen locale.
This file contains the default format ANTLR uses.

```
*/
```


location(file, line, column) ::= "<file>(<line>,<column>)"

message(id, text) ::= "error <id> : <text>"

report(location, message, type) ::= "<location> : <type> <message.id> : <message.text>"

wantsSingleLineMessage() ::= "true"

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-

jar/org/antlr/v4/tool/templates/messages/formats/vs2005.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* Copyright (c) 2014 Eric Vergnaud

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** ANTLR tool checks output templates are compatible with tool code generation.

* For now, a simple string match used on x.y of x.y.z scheme.

* Must match Tool.VERSION during load to templates.

*

* REQUIRED.

*/

```
pythonTypeInitMap ::= [  
    "bool":"False",  
    "int":"0",  
    "float":"0.0",  
    "str": "",  
    default:"None" // anything other than a primitive type is an object  
]
```

```
// args must be <object-model-object>, <fields-resulting-in-STs>
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
# encoding: utf-8  
from antlr4 import *  
from io import StringIO  
from typing.io import TextIO  
import sys
```

```
<namedActions.header>
```

```
<parser>
```

```
>>
```

```
ListenerFile(file, header, namedActions) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
from antlr4 import *  
if __name__ is not None and "." in __name__:  
    from <file.parserName> import <file.parserName>  
else:  
    from <file.parserName> import <file.parserName>  
<header>
```

```
# This class defines a complete listener for a parse tree produced by <file.parserName>.
```

```
class <file.grammarName>Listener(ParseTreeListener):
```

```
    <file.listenerNames>: {Iname |
```

```
    # Enter a parse tree produced by <file.parserName>#<Iname>.
```

```
    def enter<Iname>(self, ctx:<file.parserName>.<Iname>Context):  
        pass
```

```
    # Exit a parse tree produced by <file.parserName>#<Iname>.
```

```
    def exit<Iname>(self, ctx:<file.parserName>.<Iname>Context):  
        pass
```

```
}; separator="\n">
```

>>

```
VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
if __name__ is not None and "." in __name__:
    from .<file.parserName> import <file.parserName>
else:
    from <file.parserName> import <file.parserName>
<header>
```

This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

```
class <file.grammarName>Visitor(ParseTreeVisitor):
```

```
    <file.visitorNames:{lname |
# Visit a parse tree produced by <file.parserName>#<lname>.
def visit<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
    return self.visitChildren(ctx)

}; separator="\n">
```

```
del <file.parserName>
```

>>

```
fileHeader(grammarFileName, ANTLRVersion) ::= <<
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
>>
```

```
Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>
```

```
Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
    from .<superClass> import <superClass>
else:
    from <superClass> import <superClass>

<endif>
<atn>
```

```
class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):
```

```

grammarFileName = "<parser.grammarFileName>"

atn = ATNDeserializer().deserialize(serializedATN())

decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

sharedContextCache = PredictionContextCache()

literalNames = [ <parser.literalNames:{t | <t>}; null="\\"<INVALID>\\"", separator=", ", wrap, anchor> ]

symbolicNames = [ <parser.symbolicNames:{t | <t>}; null="\\"<INVALID>\\"", separator=", ", wrap, anchor> ]

<parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor>

ruleNames = [ <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ]

EOF = <TokenLabelType().EOF
<if(parser.tokens)>
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor>
<endif>

<parser:(ctor)()>

<namedActions.members>

<funcs; separator="\n">

<if( sempredFuncs)>
def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
    if self._predicates == None:
        self._predicates = dict()
<parser.sempredFuncs.values:{ f |
    self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n    ">
    pred = self._predicates.get(ruleIndex, None)
    if pred is None:
        raise Exception("No predicate with index:" + str(ruleIndex))
    else:
        return pred(localctx, predIndex)

<sempredFuncs.values; separator="\n">
<endif>

>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<

```

```

<if(actionFuncs)>
def action(self, localctx:RuleContext, ruleIndex:int, actionIndex:int):
    if self._actions is None:
        actions = dict()
<recog.actionFuncs.values: {f}
    actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
    self._actions = actions
    action = self._actions.get(ruleIndex, None)
    if action is not None:
        action(localctx, actionIndex)
    else:
        raise Exception("No registered action for:" + str(ruleIndex))

<actionFuncs.values; separator="\n">

<endif>
<if(sempredFuncs)>
def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
    if self._predicates is None:
        preds = dict()
<recog.sempredFuncs.values: {f}
    preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
    self._predicates = preds
    pred = self._predicates.get(ruleIndex, None)
    if pred is not None:
        return pred(localctx, predIndex)
    else:
        raise Exception("No registered predicate for:" + str(ruleIndex))

<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
def __init__(self, input:TokenStream, output:TextIO = sys.stdout):
    super().__init__(input, output)
    self.checkVersion("<file.ANTLRVersion>")
    self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
    self._predicates = None

>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

def <r.name>_action(self, localctx:<r.ctxType> , actionIndex:int):

```

```

<actions:{index|
<if(first(actions))>
  if actionIndex == <index>:
    <actions.(index)>
<elseif(rest(actions))>
  elif actionIndex == <index>:
    <actions.(index)>
<endif> }; separator="\n">
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
def <r.name>_sempred(self, localctx:<r.ctxType>, predIndex:int):
  <actions:{index|
<if(first(actions))>
  if predIndex == <index>:
    return <actions.(index)>
<elseif(rest(actions))>
  elif predIndex == <index>:
    return <actions.(index)>
<endif> }; separator="\n">
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self<currentRule.args:{a | , <a.name><if(a.type)>:<a.type><endif> }>):

  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args:{a | , <a.name> }>)
  self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
  <namedActions.init>
  <locals; separator="\n">
  try:
    <code>
    <postamble; separator="\n">
    <namedActions.after>
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  except RecognitionException as re:
    localctx.exception = re

```

```

        self._errHandler.reportError(self, re)
        self._errHandler.recover(self, re)
    <endif>
    finally:
        <finallyAction>
        self.exitRule()
    return localctx

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self, _p:int=0<if(currentRule.args)>, <args:{ a | , <a>}><endif>):
    _parentctx = self._ctx
    _parentState = self.state
    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{ a | , <a.name>}>)>
    _prevctx = localctx
    _startState = <currentRule.startState>
    self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
    <namedActions.init>
    <locals; separator="\n">
    try:
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    except RecognitionException as re:
        localctx.exception = re
        self._errHandler.reportError(self, re)
        self._errHandler.recover(self, re)
    finally:
        <finallyAction>
        self.unrollRecursionContexts(_parentctx)
    return localctx

>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel>
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\n1">
else:
  <error>

>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\n1">
else:
  pass

>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
  <alts; separator="\n">

<!else if ( !(<followExpr> ) ) <error!>

>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
  <alts; separator="\n">

```



```

self.state = <choice.loopBackStateNumber>
self._errHandler.sync(self)
<iteration>

>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">
while True:
    <alts; separator="\n">
    self.state = <choice.stateNumber> <! loopback/exit decision !>
    self._errHandler.sync(self)
    <iteration>
    if not (<loopExpr>):
        break

>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{ alt |
if la_ == <i>:
    <alt>
    pass
}; separator="\nel">

>>

OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{ alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
}; separator="\nel">

>>

StarBlock(choice, alts, sync, iteration) ::= <<

```

```

self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
    if _alt==1<if(!choice.ast.greedy)>+1<endif>:
        <iteration>
        <alts> <! should only be one !>
        self.state = <choice.loopBackStateNumber>
        self._errHandler.sync(self)
        _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
>>

```

```

PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
    <alts:{alt|
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
}; separator="\nел">
    else:
        <error>
        self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
        self._errHandler.sync(self)
        _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
>>

```

```

Sync(s) ::= "sync(<s.expecting.name>)"

```

```

ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"

```

```

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" or ">
>>

```

```

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

```

```

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> and ((1 \<< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<< <offsetShiftType(ttype, bits.shift)>)); separator=" | ">)) != 0)

```

```

%>

isZero ::= [
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ ttype | <s.varName>==<parser.name>.<ttype> }; separator=" or ">
%>

cases(ttypes) ::= <<
if token in [<ttypes:{ t | <parser.name>.<t> }; separator=", ">]:
>>

InvokeRule(r, argExprsChunks) ::= <<
self.state = <r.stateNumber>
<if(r.labels)><r.labels:{1 | <labelref(1)> =
}><endif>self.<r.name><(if(r.ast.options.p)><r.ast.options.p><(if(argExprsChunks)><endif><endif><argExprsChu
nks)>
>>

MatchToken(m) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{1 | <labelref(1)> = }><endif>self.match(<parser.name>.<m.name>)
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{1 | <labelref(1)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr><endif>:
<if(m.labels)><m.labels:{1 | <labelref(1)> = }><else> <endif>self._errHandler.recoverInline(self)
else:
self._errHandler.reportMatch(self)

```

```

    self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels:{1 | <labelref(1)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:
    from antlr4.error.Errors import FailedPredicateException
    raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"
LexerMoreCommand() ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg, grammar) ::= "_type = <arg>"
LexerChannelCommand(arg, grammar) ::= "_channel = <arg>"
LexerModeCommand(arg, grammar) ::= "_mode = <arg>"
LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValRef(a) ::= "localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"

```

```

ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type()"
TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(0 if (<ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r) ::= "(None if <ctx(r)>.<r.label> is None else
self._input.getText((<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r) ::= "localctx.stop"
ThisRulePropertyRef_text(r) ::= "self._input.getText((localctx.start, self._input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "localctx"
ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"

ContextTokenGetterDecl(t) ::= <<
def <t.name>(self):
    return self.getToken(<parser.name>.<t.name>, 0)
>>

// should never be called
ContextTokenListGetterDecl(t) ::= <<
def <t.name>_list(self):
    return self.getTokens(<parser.name>.<t.name>)
>>

```

```

ContextTokenListIndexedGetterDecl(t) ::= <<
def <t.name>(self, i:int=None):
  if i is None:
    return self.getTokens(<parser.name>.<t.name>)
  else:
    return self.getToken(<parser.name>.<t.name>, i)
>>

ContextRuleGetterDecl(r) ::= <<
def <r.name>(self):
  return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)

>>

// should never be called
ContextRuleListGetterDecl(r) ::= <<
def <r.name>_list(self):
  return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r) ::= <<
def <r.name>(self, i:int=None):
  if i is None:
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
  else:
    return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)

>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):

```

```

def __init__(self, parser, parent:ParserRuleContext=None, invokingState:int=-1<struct.ctorAttrs:{ a | ,
<a.name><if(a.type)>:<a.type><endif>=None }>):
    super().__init__(parent, invokingState)
    self.parser = parser
    <attrs:{ a | <a> }; separator="\n">
    <struct.ctorAttrs:{ a | self.<a.name> = <a.name> }; separator="\n">

<getters:{ g | <g> }; separator="\n\n">

def getRuleIndex(self):
    return <parser.name>.RULE_<struct.derivedFromName>

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
def copyFrom(self, ctx:ParserRuleContext):
    super().copyFrom(ctx)
    <struct.attrs:{ a | self.<a.name> = ctx.<a.name> }; separator="\n">

<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
class <struct.name><(currentRule.name; format="cap">Context):

    def __init__(self, parser, ctx:ParserRuleContext): # actually a <parser.name>.<currentRule.name;
format="cap">Context
        super().__init__(parser)
        <attrs:{ a | <a> }; separator="\n">
        self.copyFrom(ctx)

    <getters:{ g | <g> }; separator="\n">

    <dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener:ParseTreeListener):
    if hasattr( listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">" ):
        listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor:ParseTreeVisitor):

```

```

if hasattr( visitor, "visit<struct.derivedFromName; format="cap">" ):
    return visitor.visit<struct.derivedFromName; format="cap">(self)
else:
    return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "self._ctx.stop = self._input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName>;
format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
    self.triggerExitRuleEvent()

```



```
_prevctx = localctx
```

```
>>
```

```
LexerFile(lexerFile, lexer, namedActions) ::= <<  
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>  
from antlr4 import *  
from io import StringIO  
from typing.io import TextIO  
import sys
```

```
<namedActions.header>
```

```
<lexer>
```

```
>>
```

```
Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<  
<if(superClass)>  
if __name__ is not None and "." in __name__:  
    from .<superClass> import <superClass>  
else:  
    from <superClass> import <superClass>
```

```
<endif>
```

```
<atn>
```

```
class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):
```

```
    atn = ATNDeserializer().deserialize(serializedATN())
```

```
    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]
```

```
<if(lexer.channels)>
```

```
    <lexer.channels: {c | <c> = <lexer.channels.(c)>}; separator="\n">
```

```
<endif>
```

```
<if(rest(lexer.modes))>
```

```
    <rest(lexer.modes): {m | <m> = <i>}; separator="\n">
```

```
<endif>
```

```
    <lexer.tokens: {k | <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>
```

```
    channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels: {c |  
u"<c>"}; separator=", ", wrap, anchor><endif> ]
```

```
    modeNames = [ <lexer.modes: {m | "<m>"}; separator=", ", wrap, anchor> ]
```

```
    literalNames = [ "\<INVALID>",
```

```

    <lexer.literalNames:{t | <t>}; separator=", ", wrap, anchor> ]

symbolicNames = [ "\<INVALID>",
    <lexer.symbolicNames:{t | <t>}; separator=", ", wrap, anchor> ]

ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ]

grammarFileName = "<lexer.grammarFileName>"

def __init__(self, input=None, output:TextIO = sys.stdout):
    super().__init__(input, output)
    self.checkVersion("<lexerFile.ANTLRVersion>")
    self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
    self._actions = None
    self._predicates = None

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

def serializedATN():
    with StringIO() as buf:
        buf.write("<model.serialized; wrap={ }<\n>    buf.write(">")")
        return buf.getvalue()

>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<pythonTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".py"

Found in path(s):
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
jar/org/antlr/v4/tool/templates/codegen/Python3/Python3.stg
No license file was found, but licenses were detected in source scan.

/*
 * [The "BSD license"]

```

- * Copyright (c) 2012-2016 Terence Parr
- * Copyright (c) 2012-2016 Sam Harwell
- * Copyright (c) 2015 Janyou
- * All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- *
- * 1. Redistributions of source code must retain the above copyright
- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in the
- * documentation and/or other materials provided with the distribution.
- * 3. The name of the author may not be used to endorse or promote products
- * derived from this software without specific prior written permission.
- *
- * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
- * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
- * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
- * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
- * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
- * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
- * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
- * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
- * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
- * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- */

```

SwiftTypeInitMap ::= [
  "Int": "0",
  "Int64": "0",
  "Float": "0.0",
  "Double": "0.0",
  "Bool": "false",
  default: "nil" // anything other than a primitive type is an object
]
SwiftTypeMap ::= [
  "int": "Int",
  "float": "Float",
  "long": "Int64",
  "double": "Double",
  "bool": "Bool",
  "boolean": "Bool",
  default : key
]
// args must be <object-model-object>, <fields-resulting-in-STs>

```

```
accessLevelOpenOK(obj) ::= "<obj.accessLevel; null=\"open\">"
accessLevelNotOpen(obj) ::= "<obj.accessLevel; null=\"public\">"
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<namedActions.header>
import Antlr4

<parser>
>>
```

```
ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4
```

```
/**
 * This interface defines a complete listener for a parse tree produced by
 * { @link <file.parserName> }.
 */
<accessLevelNotOpen(file)> protocol <file.grammarName>Listener: ParseTreeListener {
  <file.listenerNames: {lname |
/**
  <if(file.listenerLabelRuleNames.(lname))>
  * Enter a parse tree produced by the { @code <lname> \}
  * labeled alternative in { @link <file.parserName> #<file.listenerLabelRuleNames.(lname)> \}.
  <else>
  * Enter a parse tree produced by { @link <file.parserName> #<lname> \}.
  <endif>
  - Parameters:
    - ctx: the parse tree
  */
  func enter<lname; format="cap">(_ ctx: <file.parserName>.<lname; format="cap">Context)
/**
  <if(file.listenerLabelRuleNames.(lname))>
  * Exit a parse tree produced by the { @code <lname> \}
  * labeled alternative in { @link <file.parserName> #<file.listenerLabelRuleNames.(lname)> \}.
  <else>
  * Exit a parse tree produced by { @link <file.parserName> #<lname> \}.
  <endif>
  - Parameters:
    - ctx: the parse tree
```

```

*/
func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname; format="cap">Context)); separator="\n">
}
>>

```

```

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>

```

```

import Antlr4

```

```

/**
 * This class provides an empty implementation of { @link <file.grammarName>Listener},
 * which can be extended to create a listener which only needs to handle a subset
 * of the available methods.
 */
<accessLevelOpenOK(file)> class <file.grammarName>BaseListener: <file.grammarName>Listener {
    <accessLevelNotOpen(file)> init() { \}
    <file.listenerNames: {lname |

```

```

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func enter<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \}

```

```

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \} }; separator="\n">

```

```

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func enterEveryRule(_ ctx: ParserRuleContext) { }
/**
 * { @inheritDoc\}
 *

```

```

* \<p>The default implementation does nothing.\</p>
*/
<accessLevelOpenOK(file)> func exitEveryRule(_ ctx: ParserRuleContext) { }
/**
* { @inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
<accessLevelOpenOK(file)> func visitTerminal(_ node: TerminalNode) { }
/**
* { @inheritDoc\}
*
* \<p>The default implementation does nothing.\</p>
*/
<accessLevelOpenOK(file)> func visitErrorNode(_ node: ErrorNode) { }
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
* This interface defines a complete generic visitor for a parse tree produced
* by { @link <file.parserName> }.
*
* @param \<T> The return type of the visit operation. Use { @link Void } for
* operations with no return type.
*/
<accessLevelOpenOK(file)> class <file.grammarName>Visitor\<T>: ParseTreeVisitor\<T> {
<file.visitorNames:{Iname |
/**
<if(file.visitorLabelRuleNames.(Iname))>
* Visit a parse tree produced by the { @code <Iname>}
* labeled alternative in { @link <file.parserName>#<file.visitorLabelRuleNames.(Iname)> }.
<else>
* Visit a parse tree produced by { @link <file.parserName>#<Iname> }.
<endif>
- Parameters:
- ctx: the parse tree
- returns: the visitor result
*/
<accessLevelOpenOK(file)> func visit<Iname; format="cap">(_ ctx: <file.parserName>.<Iname;
format="cap">Context) -> T {

```

```

fatalError(#function + " must be overridden")
\}
}; separator="\n">
}
>>

```

```

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
 * This class provides an empty implementation of { @link <file.grammarName>Visitor},
 * which can be extended to create a visitor which only needs to handle a subset
 * of the available methods.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void} for
 * operations with no return type.
 */
<accessLevelOpenOK(file)> class <file.grammarName>BaseVisitor\<T>: AbstractParseTreeVisitor\<T> {
  <file.visitorNames:{ lname |
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation returns the result of calling
 * { @link #visitChildren\} on { @code ctx\}.\</p>
 */
<accessLevelOpenOK(file)> func visit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) -> T? { return visitChildren(ctx) \}}; separator="\n">
}
>>

```

```

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

```

```

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

```

```

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<!!@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})!>
<accessLevelOpenOK(parser)> class <parser.name>: <superClass; null="Parser"> {

```

```

  internal static var _decisionToDFA: [DFA] = {

```

```

    var decisionToDFA = [DFA]()
    let length = <parser.name>._ATN.getNumberOfDecisions()
    for i in 0..

```



```

switch (ruleIndex) {
  <parser.sempredFuncs.values:{f}
case <f.ruleIndex>:
return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)); separator="\n">
  default: return true
}
<!return true;!>
}
<sempredFuncs.values; separator="\n">
<endif>

```

```
<atn>
```

```
<accessLevelNotOpen(parser)>
```

```
static let _serializedATN = <parser.name>ATN().jsonString
```

```
<accessLevelNotOpen(parser)>
```

```
static let _ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
```

```
}
```

```
>>
```

```
vocabulary(literalNames, symbolicNames, accessLevel) ::= <<
```

```
private static let _LITERAL_NAMES: [String?] = [
```

```
<literalNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
```

```
]
```

```
private static let _SYMBOLIC_NAMES: [String?] = [
```

```
<symbolicNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
```

```
]
```

```
<accessLevel>
```

```
static let VOCABULARY = Vocabulary(_LITERAL_NAMES, _SYMBOLIC_NAMES)
```

```
>>
```

```
dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
```

```
<if(actionFuncs)>
```

```
override <accessLevelOpenOK(parser)>
```

```
func action(_ _localctx: RuleContext?, _ ruleIndex: Int, _ actionIndex: Int) throws {
```

```
switch (ruleIndex) {
```

```
<recog.actionFuncs.values:{f}
```

```
case <f.ruleIndex>:
```

```
<f.name>_action((_localctx as <f.ctxType>?), actionIndex)
```

```
}; separator="\n">
```

```
default: break
```

```
}
```

```
}
```

```
<actionFuncs.values; separator="\n">
```

```
<endif>
```

```
<if(sempredFuncs)>
```

```
override <accessLevelOpenOK(parser)>
```

```

func sempred(_ _localctx:RuleContext?, _ ruleIndex: Int, _ predIndex: Int) throws -> Bool {
  switch (ruleIndex) {
    <recog.sempredFuncs.values: {f|
  case <f.ruleIndex>:
    return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)); separator="\n">
  default: return true
  }
  <!return true;!>
}
<sempredFuncs.values; separator="\n">
<endif>
>>

```

```

parser_ctor(p) ::= <<

```

```

override <accessLevelOpenOK(parser)>
func getVocabulary() -> Vocabulary {
  return <p.name>.VOCABULARY
}

```

```

override <accessLevelNotOpen(parser)>
init(_ input:TokenStream) throws {
  RuntimeMetaData.checkVersion("4.7.2", RuntimeMetaData.VERSION)
  try super.init(input)
  _interp = ParserATNSimulator(self,<p.name>._ATN,<p.name>._decisionToDFA,
<p.name>._sharedContextCache)
}

```

```

>>

```

```

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleActionFunction(r, actions) ::= <<
private func <r.name>_action(_ _localctx: <r.ctxType>?, _ actionIndex: Int) {
  switch (actionIndex) {
    <actions: {index|
  case <index>:
    <actions.(index)>
    }; separator="\n">
  default: break
  }
}
>>

```

```

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleSempredFunction(r, actions) ::= <<
private func <r.name>_sempred(_localctx: <r.ctxType>!, _predIndex: Int) throws -> Bool {
switch (predIndex) {
<actions:{index|
case <index>:return <actions.(index)>; separator="\n">
default: return true
}
<!return true;!>
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<
<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>; separator="\n">
@discardableResult
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else> <accessLevelOpenOK(parser)> func
<endif><currentRule.name>(<if(first(args))>_<endif><args; separator=", _">) throws -> <currentRule.ctxType> {
var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, getState()<currentRule.args:{a | ,
<a.name>}>)
try enterRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>)
<namedActions.init>
<locals; separator="\n">
defer {
<finallyAction>
try! exitRule()
}
do {
<if(currentRule.hasLookaheadBlock)>
var _alt:Int
<endif>
<code>
<postamble; separator="\n">
<namedActions.after>
}
<if(exceptions)>
<exceptions; separator="\n">
<else>
catch ANTLRException.recognition(let re) {
_localctx.exception = re
_errHandler.reportError(self, re)
try _errHandler.recover(self, re)
}
<endif>

return _localctx
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else> <accessLevelNotOpen(parser)> final <endif>
func <currentRule.name>( <if(first(args))>_ <endif><args; separator=", _">) throws -> <currentRule.ctxType> {
    return try <currentRule.name>(0<currentRule.args:{a | , <a.name>}>)
}
@discardableResult
private func <currentRule.name>(_ _p<args:{a | , <a>}>: Int) throws -> <currentRule.ctxType> {
    let _parentctx: ParserRuleContext? = _ctx
    var _parentState: Int = getState()
    var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
<a.name>}>)
    var _prevctx: <currentRule.ctxType> = _localctx
    var _startState: Int = <currentRule.startState>
    try enterRecursionRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p)
    <namedActions.init>
    <locals; separator="\n">
    defer {
        <finallyAction>
        try! unrollRecursionContexts(_parentctx)
    }
    do {
<if(currentRule.hasLookaheadBlock)>
        var _alt: Int
    <endif>
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    }
    catch ANTLRException.recognition(let re) {
        _localctx.exception = re
        _errHandler.reportError(self, re)
        try _errHandler.recover(self, re)
    }

    return _localctx;
}
>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>

```

```

try enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<!/{!>
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
<!/{!>
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
<preamble; separator="\n">
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))) {
<choice.altLook,alts:{look,alt | <cases(ttypes=look)>
<alt>
break }; separator="\n">
default:
<error>
}
>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
<alt>
break }; separator="\n">
default:
break
}
>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<preamble; separator="\n">
if (<expr>) {
<alts; separator="\n">
}
<!else if ( !(<followExpr> ) <error>!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
  setState(<choice.stateNumber>)
  try _errHandler.sync(self)
  <preamble; separator="\n">
  while (<loopExpr>) {
    <alts; separator="\n">
    setState(<choice.loopBackStateNumber>)
    try _errHandler.sync(self)
    <iteration>
  }
>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
  setState(<choice.blockStartStateNumber>) <! alt block decision !>
  try _errHandler.sync(self)
  <preamble; separator="\n">
  repeat {
    <alts; separator="\n">
    setState(<choice.stateNumber>); <! loopback/exit decision !>
    try _errHandler.sync(self)
    <iteration>
  } while (<loopExpr>)
>>

```

```
// LL(*) stuff
```

```

AltBlock(choice, preamble, alts, error) ::= <<
  setState(<choice.stateNumber>)
  try _errHandler.sync(self)
  <if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
  <preamble; separator="\n">
  switch(try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)) {
  <alts:{ alt |
  case <i>:
    <alt>
    break }; separator="\n">
  default: break
  }
>>

```

```

OptionalBlock(choice, alts, error) ::= <<
  setState(<choice.stateNumber>)
  try _errHandler.sync(self)
  switch (try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)) {
  <alts:{ alt |
  case <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
    break }; separator="\n">

```

```
default: break
```

```
}
```

```
>>
```

```
StarBlock(choice, alts, sync, iteration) ::= <<
```

```
  setState(<choice.stateNumber>)
```

```
  try _errHandler.sync(self)
```

```
  _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
```

```
  while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER) {
```

```
    if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
```

```
      <iteration>
```

```
      <alts> <! should only be one !>
```

```
    }
```

```
    setState(<choice.loopBackStateNumber>)
```

```
    try _errHandler.sync(self)
```

```
    _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
```

```
  }
```

```
>>
```

```
PlusBlock(choice, alts, error) ::= <<
```

```
  setState(<choice.blockStartStateNumber>); <! alt block decision !>
```

```
  try _errHandler.sync(self)
```

```
  _alt = 1<if(!choice.ast.greedy)>+1<endif>;
```

```
  repeat {
```

```
    switch (_alt) {
```

```
      <alts>:{alt|
```

```
      case <i><if(!choice.ast.greedy)>+1<endif>:
```

```
        <alt>
```

```
        break }; separator="\n">
```

```
      default:
```

```
        <error>
```

```
    }
```

```
    setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
```

```
    try _errHandler.sync(self)
```

```
    _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)
```

```
  } while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER)
```

```
>>
```

```
Sync(s) ::= "sync(<s.expecting.name>);"
```

```
ThrowNoViableAlt(t) ::= "throw ANTLRException.recognition(e: NoViableAltException(self))"
```

```
TestSetInline(s) ::= <<
```

```
  <!<s.bitsets: {bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s, bits)><endif>}; separator=" || ">!>
```

```
  //closure
```

```
  { () -> Bool in
```

```
    <if(rest(s.bitsets)>>var<else>let<endif> testSet: Bool = <first(s.bitsets):{bits |
```

```

<if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s, bits)><endif>>
  <rest(s.bitsets):{bits | testSet = testSet || <if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s,
bits)><else><bitsetInlineComparison(s, bits)><endif>>; separator="\n">
  return testSet
}()
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <<
<!(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1 \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ ttype | (1 \<< <offsetShift(ttype, bits.shift)>); separator=" | ">)) != 0)!>
{ () -> Bool in
  <! let test: Bool = (<testShiftInRange({<offsetShift(s.varName, bits.shift)>})>)!>
  <!var temp: Int64 = Int64(<offsetShift(s.varName, bits.shift)>)!>
  <!temp = (temp \< 0) ? (64 + (temp % 64)) : (temp % 64)!>
  <!let test1: Int64 = (Int64(1) \<< temp)!>
  <!var test2: Int64 = (<first(bits.ttypes):{ ttype | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)}>)!>
  <!<rest(bits.ttypes):{ ttype | test2 = test2 | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)}>;
separator="\n">!>
  let testArray: [Int] = [<s.varName>, <bits.ttypes:{ ttype |<parserName(ttype)>}; separator=", ">]
  <!var test2: Int64 = Utils.testBitLeftShiftArray(testArray)!>
  return Utils.testBitLeftShiftArray(testArray, <bits.shift>)
}()
>>

isZero ::= [
"0": true,
default: false
]
parserName(ttype) ::= <%
<parser.name>.Tokens.<ttype>.rawValue
%>
offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ ttype | <s.varName> == <parser.name>.Tokens.<ttype>.rawValue}; separator=" || ">
%>

cases(ttypes) ::= <<
<trunc(ttypes): { t | case .<t>:fallthrough } ; separator="\n">

```



```
<last(ttypes): {t | case .<t>:} ; separator="\n">
```

```
>>
```

```
InvokeRule(r, argExprsChunks) ::= <<
```

```
setState(<r.stateNumber>)
```

```
<if(r.labels)>
```

```
try {
```

```
  let assignmentValue = try
```

```
<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)
```

```
  <r.labels:{l | <labelref(l)> = assignmentValue} ; separator="\n">
```

```
  }()
```

```
<else>try
```

```
<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)<endif>
```

```
>>
```

```
MatchToken(m) ::= <<
```

```
setState(<m.stateNumber>)
```

```
<if(m.labels)>
```

```
try {
```

```
  let assignmentValue = try match(<parser.name>.Tokens.<m.name>.rawValue)
```

```
  <m.labels:{l | <labelref(l)> = assignmentValue} ; separator="\n">
```

```
  }()
```

```
<else>try match(<parser.name>.Tokens.<m.name>.rawValue)<endif>
```

```
>>
```

```
MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"
```

```
MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"
```

```
CommonSetStuff(m, expr, capture, invert) ::= <<
```

```
setState(<m.stateNumber>)
```

```
<if(m.labels)><m.labels:{l | <labelref(l)> = }>try _input.LT(1)<endif>
```

```
<capture>
```

```
if (<if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>)) {
```

```
  <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>try _errHandler.recoverInline(self)<if(m.labels)> as
```

```
Token<endif>
```

```
}
```

```
else {
```

```
  _errHandler.reportMatch(self)
```

```
  try consume()
```

```
}
```

```
>>
```

```
Wildcard(w) ::= <<
```

```
setState(<w.stateNumber>)
```

```
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>try matchWildcard();
```

```
>>
```

```

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
setState(<p.stateNumber>)
if (!(<chunks>)) {
    throw ANTLRException.recognition(e:FailedPredicateException(self, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>))
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"
LexerMoreCommand() ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg) ::= "_type = <arg>"
LexerChannelCommand(arg) ::= "_channel = <arg>"
LexerModeCommand(arg) ::= "_mode = <arg>"
LexerPushModeCommand(arg) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValRef(a) ::= "_localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getText()! : \"\""
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getType() : 0)"

```

```

TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getLine() : 0)"
TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getCharPositionInLine() : 0)"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getChannel() : 0)"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getTokenIndex() : 0)"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label> != nil ? Int(<ctx(t)>.<t.label>!.getText(!) : 0)"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.start?.description ?? \"\") : \"\""
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.stop?.description ?? \"\") : \"\""
RulePropertyRef_text(r) ::= "<ctx(r)>.<r.label> != nil ? try
_input.getText(<ctx(r)>.<r.label>!.start,<ctx(r)>.<r.label>!.stop) : \"\""
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r) ::= "_localctx.stop"
ThisRulePropertyRef_text(r) ::= "(try _input.getText(_localctx.start, try _input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "_localctx"
ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "(<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::=
"(<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "<t.name>: <SwiftTypeMap.(TokenLabelType())>!"
TokenTypeDecl(t) ::= "var <t.name>: Int = 0"
TokenListDecl(t) ::= "<t.name>: [Token] = [Token]()"
RuleContextDecl(r) ::= "<r.name>: <r.ctxName>!"
RuleContextListDecl(rdecl) ::= "<rdecl.name>: [<rdecl.ctxName>] = [<rdecl.ctxName>]()"

ContextTokenGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>() -> TerminalNode? {
return getToken(<parser.name>.Tokens.<t.name>.rawValue, 0)
}
>>

ContextTokenListGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>() -> [TerminalNode] {
return getTokens(<parser.name>.Tokens.<t.name>.rawValue)
}
>>

ContextTokenListIndexedGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>(_ i:Int) -> TerminalNode? {
return getToken(<parser.name>.Tokens.<t.name>.rawValue, i)
}

```

```

>>
ContextRuleGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>() -> <r.ctxName>? {
    return getRuleContext(<r.ctxName>.self, 0)
  }
>>

ContextRuleListGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>() -> [<r.ctxName>] {
    return getRuleContexts(<r.ctxName>.self)
  }
>>

ContextRuleListIndexedGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>(_ i: Int) -> <r.ctxName>? {
    return getRuleContext(<r.ctxName>.self, i)
  }
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = try _input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = try _input.LA(1)"

StructDecl(struct,ctorAttrs,attrs,getters,dispatchMethods,interfaces,extensionMembers,
  superClass={ParserRuleContext}) ::= <<

<accessLevelNotOpen(parser)> class <struct.name>:
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces>
separator=", "><endif> {
  <attrs:{a | <accessLevelOpenOK(parser)> var <a>}; separator="\n">
  <getters:{g | <g>}; separator="\n">
  <! <if(ctorAttrs)> <accessLevelNotOpen(parser)> init(_ parent: ParserRuleContext,_ invokingState: Int) {
super.init(parent, invokingState) }<endif> !>
  <if(ctorAttrs)>
  <accessLevelNotOpen(parser)> convenience init(_ parent: ParserRuleContext?, _ invokingState: Int<ctorAttrs:{a | ,
_ <a>}>) {

```

```

self.init(parent, invokingState)
<struct.ctorAttrs:{a | self.<a.name> = <a.name>;}; separator="\n">
}
<endif>

override <accessLevelOpenOK(parser)>
func getRuleIndex() -> Int {
return <parser.name>.RULE_<struct.derivedFromName>
}
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
<! <accessLevelNotOpen(parser)> init() { }!>
<accessLevelOpenOK(parser)>
func copyFrom(_ ctx: <struct.name>) {
super.copyFrom(ctx)
<struct.attrs:{a | self.<a.name> = ctx.<a.name>;}; separator="\n">
}
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">
}
>>

```

```

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
<accessLevelNotOpen(parser)> class <struct.name>: <currentRule.name; format="cap">Context {
<attrs:{a | <accessLevelNotOpen(parser)> var <a>;}; separator="\n">
<getters:{g | <g>;}; separator="\n">

<accessLevelNotOpen(parser)>
init(_ ctx: <currentRule.name; format="cap">Context) {
super.init()
copyFrom(ctx)
}
<dispatchMethods; separator="\n">
}
>>

```

```

ListenerDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func <if(method.isEnter)>enter<else>exit<endif>Rule(_ listener: ParseTreeListener) {
if let listener = listener as? <parser.grammarName>Listener {
listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)
}
}
>>

```

```

VisitorDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func accept<T>(_ visitor: ParseTreeVisitor<T>) -> T? {

```

```

if let visitor = visitor as? <parser.grammarName>Visitor {
    return visitor.visit<struct.derivedFromName; format="cap">(self)
}
else if let visitor = visitor as? <parser.grammarName>BaseVisitor {
    return visitor.visit<struct.derivedFromName; format="cap">(self)
}
else {
    return visitor.visitChildren(self)
}
}
>>

```

```
AttributeDecl(d) ::= "<d.name>: <SwiftTypeMap.(d.type)><if(d.initValue)> = <d.initValue><else>!<endif>"
```

```
/** If we don't know location of label def x, use this template (_localctx as! <x.ctx.name> */
```

```
labelref(x) ::= "<if(!x.isLocal)>_localctx.castdown(<x.ctx.name>.self).<endif><x.name>"
```

```
/** For any action chunk, what is correctly-typed context struct ptr? */
```

```
ctx(actionChunk) ::= "_localctx.castdown(<actionChunk.ctx.name>.self)"
```

```
// used for left-recursive rules
```

```
recRuleAltPredicate(ruleName,opPrec) ::= "precpred(_ctx, <opPrec>)"
```

```
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
```

```
recRuleSetStopToken() ::= "_ctx!.stop = try _input.LT(-1)"
```

```
recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
```

```
_localctx = <ctxName>Context(_parentctx, _parentState);
```

```
<if(label)>
```

```
<if(isListLabel)>
```

```
_localctx.<label>.append(_prevctx)
```

```
<else>
```

```
_localctx.<label> = _prevctx
```

```
<endif>
```

```
<endif>
```

```
<if(label)>_localctx.<label> = _prevctx;<endif>
```

```
try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)
```

```
>>
```

```
recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
```

```
_localctx = <currentAltLabel; format="cap">Context( <ruleName; format="cap">Context(_parentctx,
_parentState))
```

```
<if(label)>
```

```
<if(isListLabel)>
```

```
(_localctx as! <currentAltLabel; format="cap">Context).<label>.append(_prevctx)
```

```
<else>
```

```
(_localctx as! <currentAltLabel; format="cap">Context).<label> = _prevctx
```

```
<endif>
```

```
<endif>
```

```

try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = <ctxName>Context(_localctx)
_ctx = _localctx
_prevctx = _localctx
>>

recRuleSetPrevCtx() ::= <<
if _parseListeners != nil {
  try triggerExitRuleEvent()
}
_prevctx = _localctx
>>

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
<!package <lexerFile.genPackage>;!>
<endif>
<namedActions.header>
import Antlr4

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<accessLevelOpenOK(lexer)> class <lexer.name>: <superClass; null="Lexer"> {

  internal static var _decisionToDFA: [DFA] = {
    var decisionToDFA = [DFA]()
    let length = <lexer.name>._ATN.getNumberOfDecisions()
    for i in 0..<length> {
      <! decisionToDFA[i] = DFA(<lexer.name>._ATN.getDecisionState(i)!, i);!>
      decisionToDFA.append(DFA(<lexer.name>._ATN.getDecisionState(i)!, i))
    }
    return decisionToDFA
  }()

  internal static let _sharedContextCache = PredictionContextCache()

  <accessLevelNotOpen(lexer)>
  static let <lexer.tokens>:{k | <k>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>

  <if(lexer.channels)>
  <accessLevelNotOpen(lexer)>
  static let <lexer.channels>:{k | <k>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>

```

```

<endif>
<if(rest(lexer.modes))>
<accessLevelNotOpen(lexer)>
static let <rest(lexer.modes):{m| <m>=<i>}; separator=", ", wrap, anchor>
<endif>
<accessLevelNotOpen(lexer)>
static let channelNames: [String] = [
  "DEFAULT_TOKEN_CHANNEL", "HIDDEN" <if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
]

<accessLevelNotOpen(lexer)>
static let modeNames: [String] = [
  <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
]

<accessLevelNotOpen(lexer)>
static let ruleNames: [String] = [
  <lexer.ruleNames:{r| "<r>"}; separator=", ", wrap, anchor>
]

<vocabulary(lexer.literalNames, lexer.symbolicNames,
  accessLevelNotOpen(lexer))>

<namedActions.members>

override <accessLevelOpenOK(lexer)>
func getVocabulary() -> Vocabulary {
  return <lexer.name>.VOCABULARY
}

<accessLevelNotOpen(lexer)>
required init(_ input: CharStream) {
  RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION)
  super.init(input)
  _interp = LexerATNSimulator(self, <lexer.name>._ATN, <lexer.name>._decisionToDFA,
<lexer.name>._sharedContextCache)
}

override <accessLevelOpenOK(lexer)>
func getGrammarFileName() -> String { return "<lexer.grammarFileName>" }

override <accessLevelOpenOK(lexer)>
func getRuleNames() -> [String] { return <lexer.name>.ruleNames }

override <accessLevelOpenOK(lexer)>
func getSerializedATN() -> String { return <lexer.name>._serializedATN }

```



```
override <accessLevelOpenOK(lexer)>
func getChannelNames() -> [String] { return <lexer.name>.channelNames }
```

```
override <accessLevelOpenOK(lexer)>
func getModeNames() -> [String] { return <lexer.name>.modeNames }
```

```
override <accessLevelOpenOK(lexer)>
func getATN() -> ATN { return <lexer.name>._ATN }
```

```
<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
```

```
<accessLevelNotOpen(lexer)>
static let _serializedATN: String = <lexer.name>ATN().jsonString
```

```
<accessLevelNotOpen(lexer)>
static let _ATN: ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
}
>>
```

```
/** Don't need to define anything. The tool generates a XParserATN.swift file (and same for lexer)
 * which is referenced from static field _serializedATN. This json string is passed to
 * deserializeFromJson(). Note this is not the "serialization as array of ints" that other targets
 * do. It is more or less the output of ATNPrinter which gets read back in.
 */
```

```
SerializedATN(model) ::= <<
>>
```

```
/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
```

```
initValue(typeName) ::= <<
<SwiftTypeInitMap.(typeName)>
>>
```

```
codeFileExtension() ::= ".swift"
```

Found in path(s):

```
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
jar/org/antlr/v4/tool/templates/codegen/Swift/Swift.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2015 Dan McLaughlin, Mike Lischke
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

* modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/codegen/Cpp/Cpp.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD licence"]

Copyright (c) 2006 Kay Roepke

All rights reserved.

Redistribution and use in source and binary forms, with or without
 modification, are permitted provided that the following conditions
 are met:

1. Redistributions of source code must retain the above copyright
 notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
 notice, this list of conditions and the following disclaimer in the
 documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
 derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

This file contains the actual layout of the messages emitted by ANTLR.

This file contains the default format ANTLR uses.

*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<id> <text>"

report(location, message, type) ::= "<type><message.id>: <location> <message.text>"

wantsSingleLineMessage() ::= "false"

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-

jar/org/antlr/v4/tool/templates/messages/formats/antlr.stg

No license file was found, but licenses were detected in source scan.

~ Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/META-INF/maven/org/antlr/antlr4/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

```
/** templates used to generate make-compatible dependencies */
```

```
/** Generate "f : x, y, z" dependencies for input
* dependencies and generated files. in and out
* are File objects. For example, you can say
* <f.canonicalPath>
*/
dependencies(grammarFileName,in,out) ::= <<
<if(in)><grammarFileName>: <in; separator=", "><endif>
<out:{f | <f> : <grammarFileName>} ; separator="\n">
>>
```

Found in path(s):

```
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/depend.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

```
atn(startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph ATN {
rankdir=LR;
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
dfa(name, startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph <name> {
<if(rankdir)>rankdir=<rankdir>;<endif>
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
decision-rank(states) ::= <<
{rank=same; rankdir=TB; <states:{s | s<s>}; separator=" "; ">}
>>
```

```
edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>];
>>
```

```
action-edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>];
>>
```

```
epsilon-edge(src,label,target,arrowhead,transitionIndex,loopback=false) ::= <<
<src><if(transitionIndex)>p<transitionIndex><endif> -> <target> [fontname="Times-Italic",
label="&epsilon;"<if(loopback)>, style="dashed"<endif>];
>>
```

```
state(state, label, name, transitions) ::= <%
```

```

<name>[fontsize=11,
  label="
    <! rest(transition) tests for decision states: these nodes have a non-empty set of transitions after the first one. !>
    <if(rest(transitions))>
      {
        <! Label on the left side of the record node. !>
        <label>
        |
        <! Named ports in order on right side of record node, no display text. !>
        {<transitions:{t|\<p<i0>>}; separator="|">}}
      <else>
        <label>
      <endif>
    "
  <if(rest(transitions))>
    , shape=record, fixedsize=false
  <else>
    , shape=circle, fixedsize=true, width=.55
  <endif>
  , peripheries=1];
%>

```

```

stopstate(name,label,actionIndex,useBox) ::= <<
<name>[fontsize=11, label="<label><if(actionIndex)>,\naction:<actionIndex><endif>",
<if(useBox)>shape=polygon,sides=4,peripheries=2,fixedsize=false<else>shape=doublecircle, fixedsize=true,
width=.6<endif>];
>>

```

Found in path(s):

```

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/dot/graphs.stg
No license file was found, but licenses were detected in source scan.

```

```

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* Copyright (c) 2014 Eric Vergnaud
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.

```

```

* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** ANTLR tool checks output templates are compatible with tool code generation.
* For now, a simple string match used on x.y of x.y.z scheme.
* Must match Tool.VERSION during load to templates.
*
* REQUIRED.
*/

```

```

javascriptTypeInitMap ::= [
  "bool":"false",
  "int":"0",
  "float":"0.0",
  "str": "",
  default:"{}" // anything other than a primitive type is an object
]

```

```

// args must be <object-model-object>, <fields-resulting-in-STs>

```

```

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
var antlr4 = require('antlr4/index');
<if(file.genListener)>
var <file.grammarName>Listener = require('./<file.grammarName>Listener').<file.grammarName>Listener;
<endif>
<if(file.genVisitor)>
var <file.grammarName>Visitor = require('./<file.grammarName>Visitor').<file.grammarName>Visitor;
<endif>

```

```

<namedActions.header>
<parser>
>>

```

```

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>

```

```

var antlr4 = require('antlr4/index');

// This class defines a complete listener for a parse tree produced by <file.parserName>.
function <file.grammarName>Listener() {
  antlr4.tree.ParseTreeListener.call(this);
  return this;
}

<file.grammarName>Listener.prototype = Object.create(antlr4.tree.ParseTreeListener.prototype);
<file.grammarName>Listener.prototype.constructor = <file.grammarName>Listener;

<file.listenerNames>:{lname |
// Enter a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Listener.prototype.enter<lname; format="cap"> = function(ctx) {
\};

// Exit a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Listener.prototype.exit<lname; format="cap"> = function(ctx) {
\};

}; separator="\n">

exports.<file.grammarName>Listener = <file.grammarName>Listener;
>>

```

```

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
var antlr4 = require('antlr4/index');

// This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

function <file.grammarName>Visitor() {
  antlr4.tree.ParseTreeVisitor.call(this);
  return this;
}

<file.grammarName>Visitor.prototype = Object.create(antlr4.tree.ParseTreeVisitor.prototype);
<file.grammarName>Visitor.prototype.constructor = <file.grammarName>Visitor;

<file.visitorNames>:{lname |
// Visit a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Visitor.prototype.visit<lname; format="cap"> = function(ctx) {
  return this.visitChildren(ctx);
\};

}; separator="\n">

```



```

exports.<file.grammarName>Visitor = <file.grammarName>Visitor;
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
// jshint ignore: start
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<if(superClass)>
var <superClass> = require('./<superClass>').<superClass>;
<endif>

var grammarFileName = "<parser.grammarFileName; format="java-escape">";

<atn>

var atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);

var decisionsToDFA = atn.decisionToState.map( function(ds, index) { return new antlr4.dfa.DFA(ds, index); });

var sharedContextCache = new antlr4.PredictionContextCache();

var literalNames = [ <parser.literalNames:{t | <t>}; null="null", separator=",", wrap, anchor > ];

var symbolicNames = [ <parser.symbolicNames:{t | <t>}; null="null", separator=",", wrap, anchor > ];

var ruleNames = [ <parser.ruleNames:{r | "<r>"}; separator=",", wrap, anchor > ];

function <parser.name> (input) {
  <superClass; null="antlr4.Parser">.call(this, input);
  this._interp = new antlr4.atn.ParserATNSimulator(this, atn, decisionsToDFA, sharedContextCache);
  this.ruleNames = ruleNames;
  this.literalNames = literalNames;
  this.symbolicNames = symbolicNames;
  <namedActions.members>
  return this;
}

<parser.name>.prototype = Object.create(<superClass; null="antlr4.Parser">.prototype);
<parser.name>.prototype.constructor = <parser.name>;

Object.defineProperty(<parser.name>.prototype, "atn", {
  get : function() {
    return atn;
  }
});

```

```

<parser.name>.EOF = antlr4.Token.EOF;
<if(parser.tokens)>
<parser.tokens:{k | <parser.name>.<k> = <parser.tokens.(k)>;}; separator="\n", wrap, anchor>
<endif>

<parser.rules:{r | <parser.name>.RULE_<r.name> = <r.index>;}; separator="\n", wrap, anchor>

<funcs; separator="\n">

<if(semanticPredFuncs)>
<parser.name>.prototype.semanticPred = function(localctx, ruleIndex, predIndex) {
switch(ruleIndex) {
<parser.semanticPredFuncs.values:{f | case <f.ruleIndex>:
return this.<f.name>_semanticPred(localctx, predIndex);}; separator="\n">
default:
    throw "No predicate with index:" + ruleIndex;
}
};

<semanticPredFuncs.values; separator="\n">
<endif>

exports.<parser.name> = <parser.name>;

>>

dumpActions(recog, argFuncs, actionFuncs, semanticPredFuncs) ::= <<
<if(actionFuncs)>
<lexer.name>.prototype.action = function(localctx, ruleIndex, actionIndex) {
switch (ruleIndex) {
<recog.actionFuncs.values:{f|
case <f.ruleIndex>:
this.<f.name>_action(localctx, actionIndex);
break;}; separator="\n">
default:
    throw "No registered action for:" + ruleIndex;
}
};

<actionFuncs.values; separator="\n">
<endif>
<if(semanticPredFuncs)>
<lexer.name>.prototype.semanticPred = function(localctx, ruleIndex, predIndex) {
switch (ruleIndex) {
<recog.semanticPredFuncs.values:{f| case <f.ruleIndex>:
return this.<f.name>_semanticPred(localctx, predIndex);}; separator="\n">
default:

```

```

        throw "No registered predicate for:" + ruleIndex;
    }
};

<sempredFuncs.values; separator="\n">
<endif>
>>

/* This generates a private method since the actionIndex is generated, making an
 * overriding implementation impossible to maintain.
 */
RuleActionFunction(r, actions) ::= <<

<lexer.name>.prototype.<r.name>_action = function(localctx , actionIndex) {
    switch (actionIndex) {
    <actions:{index|
case <index>:
    <actions.(index)>
    break;}; separator="\n">
    default:
        throw "No registered action for:" + actionIndex;
    }
};
>>

/* This generates a private method since the predIndex is generated, making an
 * overriding implementation impossible to maintain.
 */
RuleSempredFunction(r, actions) ::= <<
<if (r.factory.g.lexer)><lexer.name><else><parser.name><endif>.prototype.<r.name>_sempred = function(localctx,
predIndex) {
    switch(predIndex) {
    <actions:{index| case <index>:
return <actions.(index)>;}; separator="\n">
    default:
        throw "No predicate with index:" + predIndex;
    }
};
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

<! Define fields of this parser to export the context classes !>
<parser.name>.<currentRule.ctxType> = <currentRule.ctxType>;

<parser.name>.prototype.<currentRule.name> = function(<currentRule.args:{a | <a.name>}; separator=", ">) {

    var localctx = new <currentRule.ctxType>(this, this._ctx, this.state<currentRule.args:{a | , <a.name>}>);
    this.enterRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>);
    <namedActions.init>
    <locals; separator="\n">
    try {
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    <if(exceptions)>
    <exceptions; separator="\n">
    <else>
    } catch (re) {
        if(re instanceof antlr4.error.RecognitionException) {
            localctx.exception = re;
            this._errHandler.reportError(this, re);
            this._errHandler.recover(this, re);
        } else {
            throw re;
        }
    }<endif> finally {
        <finallyAction>
        this.exitRule();
    }
    return localctx;
};

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<parser.name>.prototype.<currentRule.name> = function(_p<if(currentRule.args)>, <args:{a | , <a>}><endif>) {
    if(_p===undefined) {
        _p = 0;
    }
    var _parentctx = this._ctx;
    var _parentState = this.state;
    var localctx = new <currentRule.ctxType>(this, this._ctx, _parentState<args:{a | , <a.name>}>);

```

```

var _prevctx = localctx;
var _startState = <currentRule.startState>;
this.enterRecursionRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p);
<namedActions.init>
<locals; separator="\n">
try {
  <code>
  <postamble; separator="\n">
  <namedActions.after>
} catch( error) {
  if(error instanceof antlr4.error.RecognitionException) {
    localctx.exception = error;
    this._errHandler.reportError(this, error);
    this._errHandler.recover(this, error);
  } else {
    throw error;
  }
} finally {
  <finallyAction>
  this.unrollRecursionContexts(_parentctx)
}
return localctx;
};

```

>>

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(this, localctx);<endif>
this.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = this._input.LT(1);<endif>
<preamble; separator="\n">
switch(this._input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break; }; separator="\n">

```

```
default:
  <error>
}
>>
```

```
LL1OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
switch (this._input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break;}; separator="\n">
default:
  break;
}
>>
```

```
LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
if(<expr>) {
  <alts; separator="\n">
}
<!else if ( !(<followExpr> ) <error!>
>>
```

```
LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
while(<loopExpr>) {
  <alts; separator="\n">
  this.state = <choice.loopBackStateNumber>;
  this._errHandler.sync(this);
  <iteration>
}
>>
```

```
LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
<preamble; separator="\n">
do {
  <alts; separator="\n">
  this.state = <choice.stateNumber>; <! loopback/exit decision !>
  this._errHandler.sync(this);
  <iteration>
}
```

```

} while(<loopExpr>);
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
switch(la_) {
<alts:{alt |
case <i>:
    <alt>
    break;
}; separator="\n">
}
>>

OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
<alts:{alt |
if(la_===<i><if(!choice.ast.greedy)>+1<endif>) {
    <alt>
}; separator="\n} else ">
}
>>

StarBlock(choice, alts, sync, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx)
while(!_alt!=<choice.exitAlt> && !_alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER) {
    if(_alt===1<if(!choice.ast.greedy)>+1<endif>) {
        <iteration>
        <alts> <! should only be one !>
    }
    this.state = <choice.loopBackStateNumber>;
    this._errHandler.sync(this);
    _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
}

>>

PlusBlock(choice, alts, error) ::= <<

```

```

this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
var _alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
  switch (_alt) {
    <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
default:
<error>
}
this.state = <choice.loopBackStateNumber>; <! loopback/exit decision !>
this._errHandler.sync(this);
_alt = this._interp.adaptivePredict(this._input,<choice.decision>, this._ctx);
} while ( _alt!=<choice.exitAlt> && _alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER );
>>

```

```
Sync(s) ::= "sync(<s.expecting.name>)"
```

```
ThrowNoViableAlt(t) ::= "throw new antlr4.error.NoViableAltException(this);"
```

```

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

```

```

// Javascript language spec - shift operators are 32 bits long max
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x1f) == 0
>>

```

```

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> && ((1 \<< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<< <offsetShiftType(ttype, bits.shift)>)}); separator=" | ">)) !== 0)
%>

```

```

isZero ::= [
"0":true,
default:false
]

```

```

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

```

```
offsetShiftType(shiftAmount, offset) ::= <%
```



```

<if(!isZero.(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ ttype | <s.varName>===<parser.name>.<ttype> }; separator=" || ">
%>

cases(ttypes) ::= <<
<types:{ t | case <parser.name>.<t>: }; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
this.state = <r.stateNumber>;
<if(r.labels)><r.labels:{1 | <labelref(l)> =
}><endif>this.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChu
nks>);
>>

MatchToken(m) ::= <<
this.state = <m.stateNumber>;
<if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>this.match(<parser.name>.<m.name>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
this.state = <m.stateNumber>;
<if(m.labels)><m.labels:{1 | <labelref(l)> = }>this._input.LT(1);<endif>
<capture>
<if(invert)>if(<m.varName>\<=0 || <expr><else>if(!(<expr>))<endif> {
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>this._errHandler.recoverInline(this);
}
else {
  this._errHandler.reportMatch(this);
  this.consume();
}
>>

Wildcard(w) ::= <<
this.state = <w.stateNumber>;
<if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>this.matchWildcard();
>>

// ACTION STUFF

```

```

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
this.state = <p.stateNumber>;
if (!( <chunks>)) {
    throw new antlr4.error.FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "this.skip()"
LexerMoreCommand() ::= "this.more()"
LexerPopModeCommand() ::= "this.popMode()"
LexerTypeCommand(arg, grammar) ::= "this._type = <arg>"
LexerChannelCommand(arg, grammar) ::= "this._channel = <arg>"
LexerModeCommand(arg, grammar) ::= "this._mode = <arg>"
LexerPushModeCommand(arg, grammar) ::= "this.pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValRef(a) ::= "localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(<ctx(t)>.<t.label>===null ? null : <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.channel)"

```

```
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label> === null ? 0 : <ctx(t)>.<t.label>.tokenIndex)"
```

```
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label> === null ? 0 : parseInt(<ctx(t)>.<t.label>.text))"
```

```
RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.start)"
```

```
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.stop)"
```

```
RulePropertyRef_text(r) ::= "<ctx(r)>.<r.label>===null ? null : this._input.getText(new  
antlr4.Interval(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
```

```
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
```

```
RulePropertyRef_parser(r) ::= "this"
```

```
ThisRulePropertyRef_start(r) ::= "localctx.start"
```

```
ThisRulePropertyRef_stop(r) ::= "localctx.stop"
```

```
ThisRulePropertyRef_text(r) ::= "this._input.getText(new antlr4.Interval(localctx.start, this._input.LT(-1)))"
```

```
ThisRulePropertyRef_ctx(r) ::= "localctx"
```

```
ThisRulePropertyRef_parser(r) ::= "this"
```

```
NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"
```

```
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"
```

```
AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.push(<labelref(a.label)>);"
```

```
TokenDecl(t) ::= "this.<t.name> = null; // <TokenLabelType(>"
```

```
TokenTypeDecl(t) ::= "var <t.name> = 0; // <TokenLabelType(> type"
```

```
TokenListDecl(t) ::= "this.<t.name> = []; // of <TokenLabelType(>s"
```

```
RuleContextDecl(r) ::= "this.<r.name> = null; // <r.ctxName>"
```

```
RuleContextListDecl(rdecl) ::= "this.<rdecl.name> = []; // of <rdecl.ctxName>s"
```

```
ContextTokenGetterDecl(t) ::= <<
```

```
<t.name> = function() {
```

```
    return this.getToken(<parser.name>.<t.name>, 0);
```

```
};
```

```
>>
```

```
// should never be called
```

```
ContextTokenListGetterDecl(t) ::= <<
```

```
def <t.name>_list(self):
```

```
    return self.getTokens(<parser.name>.<t.name>)
```

```
>>
```

```
ContextTokenListIndexedGetterDecl(t) ::= <<
```

```
<t.name> = function(i) {
```

```
    if(i===undefined) {
```

```
        i = null;
```

```
    }
```

```
    if(i===null) {
```

```
        return this.getTokens(<parser.name>.<t.name>);
```

```
    } else {
```

```
        return this.getToken(<parser.name>.<t.name>, i);
```

```

    }
};

>>

ContextRuleGetterDecl(r) ::= <<
<r.name> = function() {
    return this.getTypedRuleContext(<r.ctxName>,0);
};
>>

// should never be called
ContextRuleListGetterDecl(r) ::= <<
def <r.name>_list(self):
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r) ::= <<
<r.name> = function(i) {
    if(i===undefined) {
        i = null;
    }
    if(i===null) {
        return this.getTypedRuleContexts(<r.ctxName>);
    } else {
        return this.getTypedRuleContext(<r.ctxName>,i);
    }
};
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = this._input.LA(1);"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<
function <struct.name>(parser, parent, invokingState<struct.ctorAttrs:{a | , <a.name>}> ) {

```

```

if(parent===undefined) {
    parent = null;
}
if(invokingState===undefined || invokingState===null) {
    invokingState = -1;
}
<if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.call(this, parent,
invokingState);
    this.parser = parser;
    this.ruleIndex = <parser.name>.RULE_<struct.derivedFromName>;
    <attrs:{ a | <a>}; separator="\n">
    <struct.ctorAttrs:{ a | this.<a.name> = <a.name> || null;}; separator="\n">
    return this;
}

<struct.name>.prototype =
Object.create(<if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.prototype);
<struct.name>.prototype.constructor = <struct.name>;

<getters:{ g | <struct.name>.prototype.<g>}; separator="\n\n">

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
<struct.name>.prototype.copyFrom = function(ctx) {
<if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.prototype.copyFrom.call(this,
ctx);
    <struct.attrs:{ a | this.<a.name> = ctx.<a.name>;}; separator="\n">
};
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
function <struct.name>(parser, ctx) {
    <currentRule.name; format="cap">Context.call(this, parser);
    <attrs:{ a | <a>}; separator="\n">
    <currentRule.name; format="cap">Context.prototype.copyFrom.call(this, ctx);
    return this;
}

<struct.name>.prototype = Object.create(<currentRule.name; format="cap">Context.prototype);
<struct.name>.prototype.constructor = <struct.name>;

<! Define fields of this parser to export this struct/context class !>
<parser.name>.<struct.name> = <struct.name>;

<getters:{ g | <struct.name>.prototype.<g>}; separator="\n\n">

```

```
<dispatchMethods; separator="\n">
```

```
>>
```

```
ListenerDispatchMethod(method) ::= <<
```

```
<struct.name>.prototype.<if(method.isEnter)>enter<else>exit<endif>Rule = function(listener) {  
  if(listener instanceof <parser.grammarName>Listener ) {  
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(this);  
  }  
};
```

```
>>
```

```
VisitorDispatchMethod(method) ::= <<
```

```
<struct.name>.prototype.accept = function(visitor) {  
  if ( visitor instanceof <parser.grammarName>Visitor ) {  
    return visitor.visit<struct.derivedFromName; format="cap">(this);  
  } else {  
    return visitor.visitChildren(this);  
  }  
};
```

```
>>
```

```
AttributeDecl(d) ::= "this.<d.name> = <if(d.initValue)><d.initValue><else>null<endif>"
```

```
/** If we don't know location of label def x, use this template */
```

```
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"
```

```
/** For any action chunk, what is correctly-typed context struct ptr? */
```

```
ctx(actionChunk) ::= "localctx"
```

```
// used for left-recursive rules
```

```
recRuleAltPredicate(ruleName,opPrec) ::= "this.precpred(this._ctx, <opPrec>)"
```

```
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
```

```
recRuleSetStopToken() ::= "this._ctx.stop = this._input.LT(-1);"
```

```
recRuleAltStartAction(ruleName, ctxName, label) ::= <<
```

```
localctx = new <ctxName>Context(this, _parentctx, _parentState);
```

```
<if(label)>localctx.<label> = _prevctx;<endif>
```

```
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
```

```
>>
```

```
recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
```

```
localctx = new <currentAltLabel; format="cap">Context(this, new <ruleName; format="cap">Context(this,  
_parentctx, _parentState));
```

```
<if(label)>
```

```
<if(isListLabel)>
```

```

localctx.<label>.push(_prevctx);
<else>
localctx.<label> = _prevctx;
<endif>
<endif>
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
>>

```

```

recRuleReplaceContext(ctxName) ::= <<
localctx = new <ctxName>Context(this, localctx);
this._ctx = localctx;
_prevctx = localctx;
>>

```

```

recRuleSetPrevCtx() ::= <<
if(this._parseListeners!==null) {
    this.triggerExitRuleEvent();
}
_prevctx = localctx;
>>

```

```

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
var antlr4 = require('antlr4/index');

```

```

<namedActions.header>

```

```

<lexer>

```

```

>>

```

```

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
var <superClass> = require('./<superClass>').<superClass>;
<endif>

```

```

<atn>

```

```

var atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);

```

```

var decisionsToDFA = atn.decisionToState.map( function(ds, index) { return new antlr4.dfa.DFA(ds, index); });

```

```

function <lexer.name>(input) {
    <if(superClass)><superClass><else>antlr4.Lexer<endif>.call(this, input);
    this._interp = new antlr4.atn.LexerATNSimulator(this, atn, decisionsToDFA, new
antlr4.PredictionContextCache());
    return this;
}

```

```

}

<lexer.name>.prototype = Object.create(<if(superClass)><superClass><else>antlr4.Lexer<endif>.prototype);
<lexer.name>.prototype.constructor = <lexer.name>;

Object.defineProperty(<lexer.name>.prototype, "atn", {
  get : function() {
    return atn;
  }
});

<lexer.name>.EOF = antlr4.Token.EOF;
<lexer.tokens:{k | <lexer.name>.<k> = <lexer.tokens.(k)>; separator="\n", wrap, anchor>

<if(lexer.channels)>
<lexer.channels:{c | <lexer.name>.<c> = <lexer.channels.(c)>; separator="\n">

<endif>
<if(rest(lexer.modes))>
<rest(lexer.modes):{m | <lexer.name>.<m> = <i>; separator="\n">

<endif>
<lexer.name>.prototype.channelNames = [ "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>,
<lexer.channels:{c | "<c>"}; separator=", ", wrap, anchor><endif> ];

<lexer.name>.prototype.modeNames = [ <lexer.modes:{m | "<m>"}; separator=", ", wrap, anchor> ];

<lexer.name>.prototype.literalNames = [ <lexer.literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];

<lexer.name>.prototype.symbolicNames = [ <lexer.symbolicNames:{t | <t>}; null="null", separator=", ", wrap,
anchor> ];

<lexer.name>.prototype.ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ];

<lexer.name>.prototype.grammarFileName = "<lexer.grammarFileName>";

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

exports.<lexer.name> = <lexer.name>;

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

var serializedATN = ["<model.serialized; wrap={",<\n> " }>"].join("");

```


>>

```
/** Using a type to init value map, try to init a type; if not in table  
* must be an object, default value is "null".
```

```
*/
```

```
initValue(typeName) ::= <<
```

```
<javacriptTypeInitMap.(typeName)>
```

```
>>
```

```
codeFileExtension() ::= ".js"
```

Found in path(s):

```
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
```

```
jar/org/antlr/v4/tool/templates/codegen/JavaScript/JavaScript.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD licence"]
```

```
Copyright (c) 2006 Kay Roepke
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/*
```

This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the format that mimicks GCC output.

*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<text> [error <id>]"

report(location, message, type) ::= "<location> <type>: <message>"

wantsSingleLineMessage() ::= "true"

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/messages/formats/gnu.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* Copyright (c) 2014 Eric Vergnaud

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** ANTLR tool checks output templates are compatible with tool code generation.

* For now, a simple string match used on x.y of x.y.z scheme.

* Must match Tool.VERSION during load to templates.

*

* REQUIRED.

*/

```
pythonTypeInitMap ::= [  
  "bool":"False",  
  "int":"0",  
  "float":"0.0",  
  "str": "",  
  default:"None" // anything other than a primitive type is an object  
]
```

```
// args must be <object-model-object>, <fields-resulting-in-STs>
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
```

```
# encoding: utf-8
```

```
from __future__ import print_function
```

```
from antlr4 import *
```

```
from io import StringIO
```

```
import sys
```

```
<namedActions.header>
```

```
<parser>
```

```
>>
```

```
ListenerFile(file, header, namedActions) ::= <<
```

```
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
```

```
from antlr4 import *
```

```
<header>
```

```
# This class defines a complete listener for a parse tree produced by <file.parserName>.
```

```
class <file.grammarName>Listener(ParseTreeListener):
```

```
    <file.listenerNames>{lname |
```

```
# Enter a parse tree produced by <file.parserName>#<lname>.
```

```
def enter<lname; format="cap">(self, ctx):
```

```
    pass
```

```
# Exit a parse tree produced by <file.parserName>#<lname>.
```

```
def exit<lname; format="cap">(self, ctx):
```

```
    pass
```

```
}; separator="\n">
```

```
>>
```

```

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
<header>

# This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

class <file.grammarName>Visitor(ParseTreeVisitor):

    <file.visitorNames:{ lname |
# Visit a parse tree produced by <file.parserName>#<lname>.
def visit<lname; format="cap">(self, ctx):
    return self.visitChildren(ctx)

}; separator="\n">

>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(superClass)>
from .<superClass> import <superClass>

<endif>
<atn>

class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):

    grammarFileName = "<parser.grammarFileName>"

    atn = ATNDeserializer().deserialize(serializedATN())

    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

    sharedContextCache = PredictionContextCache()

    literalNames = [ <parser.literalNames:{ t | u<t> }; null="u\"<INVALID>\"", separator=", ", wrap, anchor ]

```

```

symbolicNames = [ <parser.symbolicNames:{t | u<t>}; null="u\"<INVALID>\"", separator=", ", wrap, anchor ]

<parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor>

ruleNames = [ <parser.ruleNames:{r | u"<r>"}; separator=", ", wrap, anchor ]

EOF = <TokenLabelType().EOF
<if(parser.tokens)>
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor>
<endif>

<parser:(ctor())>

<namedActions.members>

<funcs; separator="\n">

<if(sempredFuncs)>
def sempred(self, localctx, ruleIndex, predIndex):
    if self._predicates == None:
        self._predicates = dict()
<parser.sempredFuncs.values:{f |
    self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n    ">
    pred = self._predicates.get(ruleIndex, None)
    if pred is None:
        raise Exception("No predicate with index:" + str(ruleIndex))
    else:
        return pred(localctx, predIndex)

<sempredFuncs.values; separator="\n">
<endif>

>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
def action(self, localctx, ruleIndex, actionIndex):
    if self._actions is None:
        actions = dict()
<recog.actionFuncs.values:{f|
    actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
    self._actions = actions
    action = self._actions.get(ruleIndex, None)
    if action is not None:
        action(localctx, actionIndex)

```

```

else:
    raise Exception("No registered action for:" + str(ruleIndex))

<actionFuncs.values; separator="\n">

<endif>
<if(sempredFuncs)>
def sempred(self, localctx, ruleIndex, predIndex):
    if self._predicates is None:
        preds = dict()
    <recog.sempredFuncs.values:{f}
        preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
        self._predicates = preds
    pred = self._predicates.get(ruleIndex, None)
    if pred is not None:
        return pred(localctx, predIndex)
    else:
        raise Exception("No registered predicate for:" + str(ruleIndex))

<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
def __init__(self, input, output=sys.stdout):
    super(<parser.name>, self).__init__(input, output=output)
    self.checkVersion("<file.ANTLRVersion>")
    self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
    self._predicates = None

>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

def <r.name>_action(self, localctx , actionIndex):
<actions:{index|
<if(first(actions))>
    if actionIndex == <index>:
        <actions.(index)>
<elseif(rest(actions))>
    elif actionIndex == <index>:
        <actions.(index)>
<endif> } ; separator="\n">
>>

```

```

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
def <r.name>_sempred(self, localctx, predIndex):
    <actions: {index|
<if(first(actions))>
    if predIndex == <index>:
        return <actions.(index)>
<elseif(rest(actions))>
    elif predIndex == <index>:
        return <actions.(index)>
<endif> }; separator="\n">

>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs: {l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self<currentRule.args: {a | , <a.name>}>):

    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args: {a | , <a.name>}>)
    self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
    <namedActions.init>
    <locals; separator="\n">
    try:
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    <if(exceptions)>
    <exceptions; separator="\n">
    <else>
    except RecognitionException as re:
        localctx.exception = re
        self._errHandler.reportError(self, re)
        self._errHandler.recover(self, re)
    <endif>
    finally:
        <finallyAction>
        self.exitRule()
    return localctx

>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
  namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>} ; separator="\n">

def <currentRule.name>(self, _p=0<if(currentRule.args)>, <args:{a | , <a>}><endif>):
  _parentctx = self._ctx
  _parentState = self.state
  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{a | , <a.name>}>)
  _prevctx = localctx
  _startState = <currentRule.startState>
  self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
  <namedActions.init>
  <locals; separator="\n">
  try:
    <code>
    <postamble; separator="\n">
    <namedActions.after>
  except RecognitionException as re:
    localctx.exception = re
    self._errHandler.reportError(self, re)
    self._errHandler.recover(self, re)
  finally:
    <finallyAction>
    self.unrollRecursionContexts(_parentctx)
  return localctx

>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>

```



```

<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
    pass }; separator="\n1">
else:
  <error>

>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
    pass }; separator="\n1">
else:
  pass
>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
  <alts; separator="\n">

<!else if ( !(<followExpr> ) ) <error>!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
  <alts; separator="\n">
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  <iteration>

>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">

```

```

while True:
    <alts; separator="\n">
    self.state = <choice.stateNumber> <! loopback/exit decision !>
    self._errHandler.sync(self)
    <iteration>
    if not (<loopExpr>):
        break

>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{ alt |
if la_ == <i>:
    <alt>
    pass
}; separator="\nel">

>>

OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{ alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
}; separator="\nel">

>>

StarBlock(choice, alts, sync, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
    if _alt==1<if(!choice.ast.greedy)>+1<endif>:
        <iteration>
        <alts> <! should only be one !>
    self.state = <choice.loopBackStateNumber>
    self._errHandler.sync(self)
    _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

```

>>

```
PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  <alts:{alt|
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nел">
  else:
    <error>
self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
```

>>

```
Sync(s) ::= "sync(<s.expecting.name>)"
```

```
ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"
```

```
TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" or ">
```

>>

```
// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
```

```
((<shiftAmount>) & ~0x3f) == 0
```

>>

```
// produces smaller bytecode only when bits.ttypes contains more than two items
```

```
bitsetBitfieldComparison(s, bits) ::= <%
```

```
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> and ((1 \<< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<< <offsetShiftType(ttype, bits.shift)>)}); separator=" | ">)) != 0)
%>
```

```
isZero ::= [
"0":true,
default:false
]
```

```
offsetShiftVar(shiftAmount, offset) ::= <%
```

```
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
```

```
%>
```

```

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes: { ttype | <s.varName>==<parser.name>.<ttype> }; separator=" or ">
%>

cases(ttypes) ::= <<
if token in [<ttypes: { t | <parser.name>.<t> }; separator=", ">]:
>>

InvokeRule(r, argExprsChunks) ::= <<
self.state = <r.stateNumber>
<if(r.labels)><r.labels: { l | <labelref(l)> =
}><endif>self.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChu
nks>)
>>

MatchToken(m) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels: { l | <labelref(l)> = }><endif>self.match(<parser.name>.<m.name>)
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels: { l | <labelref(l)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr>)<endif>:
<if(m.labels)><m.labels: { l | <labelref(l)> = }><else> <endif>self._errHandler.recoverInline(self)
else:
    self._errHandler.reportMatch(self)
    self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels: { l | <labelref(l)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

```

```

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:
    from antlr4.error.Errors import FailedPredicateException
    raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"
LexerMoreCommand() ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg, grammar)    ::= "_type = <arg>"
LexerChannelCommand(arg, grammar) ::= "_channel = <arg>"
LexerModeCommand(arg, grammar)    ::= "_mode = <arg>"
LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValRef(a) ::= "localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"

```

```

TokenPropertyRef_channel(t) ::= "(0 if (<ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r) ::= "(None if <ctx(r)>.<r.label> is None else
self._input.getText((<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r) ::= "localctx.stop"
ThisRulePropertyRef_text(r) ::= "self._input.getText((localctx.start, self._input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "localctx"
ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"

ContextTokenGetterDecl(t) ::= <<
def <t.name>(self):
    return self.getToken(<parser.name>.<t.name>, 0)
>>

// should never be called
ContextTokenListGetterDecl(t) ::= <<
def <t.name>_list(self):
    return self.getTokens(<parser.name>.<t.name>)
>>

ContextTokenListIndexedGetterDecl(t) ::= <<
def <t.name>(self, i=None):
    if i is None:
        return self.getTokens(<parser.name>.<t.name>)
    else:
        return self.getToken(<parser.name>.<t.name>, i)
>>

ContextRuleGetterDecl(r) ::= <<

```

```

def <r.name>(self):
    return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)

>>

// should never be called
ContextRuleListGetterDecl(r) ::= <<
def <r.name>_list(self):
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r) ::= <<
def <r.name>(self, i=None):
    if i is None:
        return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
    else:
        return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)

>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):

    def __init__(self, parser, parent=None, invokingState=-1<struct.ctorAttrs:{a | , <a.name>=None}>>):
        super(<parser.name>.<struct.name>, self).__init__(parent, invokingState)
        self.parser = parser
        <attrs:{a | <a>}; separator="\n">
        <struct.ctorAttrs:{a | self.<a.name> = <a.name>}; separator="\n">

        <getters:{g | <g>}; separator="\n\n">

    def getRuleIndex(self):

```

```

return <parser.name>.RULE_<struct.derivedFromName>

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
def copyFrom(self, ctx):
    super(<parser.name>.<struct.name>, self).copyFrom(ctx)
    <struct.attrs:{ a | self.<a.name> = ctx.<a.name> }; separator="\n">

<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
class <struct.name>(<currentRule.name; format="cap">Context):

    def __init__(self, parser, ctx): # actually a <parser.name>.<currentRule.name; format="cap">Context)
        super(<parser.name>.<struct.name>, self).__init__(parser)
        <attrs:{ a | <a> }; separator="\n">
        self.copyFrom(ctx)

    <getters:{ g | <g> }; separator="\n">

    <dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener):
    if hasattr(listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">"):
        listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor):
    if hasattr(visitor, "visit<struct.derivedFromName; format="cap">"):
        return visitor.visit<struct.derivedFromName; format="cap">(self)
    else:
        return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

```



```

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "self._ctx.stop = self._input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName;
format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
    self.triggerExitRuleEvent()
_prevctx = localctx
>>

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
# encoding: utf-8
from __future__ import print_function
from antlr4 import *
from io import StringIO
import sys

```

```

<namedActions.header>

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
from .<superClass> import <superClass>

<endif>

<atn>

class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):

    atn = ATNDeserializer().deserialize(serializedATN())

    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

    <if(lexer.channels)>
        <lexer.channels: {c| <c> = <lexer.channels.(c)>}; separator="\n">

    <endif>
    <if(rest(lexer.modes)>
        <rest(lexer.modes): {m| <m> = <i>}; separator="\n">

    <endif>
        <lexer.tokens: {k | <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>

    channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels: {c|
u"<c>"; separator=", ", wrap, anchor><endif> ]

    modeNames = [ <lexer.modes: {m| u"<m>"; separator=", ", wrap, anchor> ]

    literalNames = [ u"\<INVALID>",
        <lexer.literalNames: {t | u<t>}; separator=", ", wrap, anchor> ]

    symbolicNames = [ u"\<INVALID>",
        <lexer.symbolicNames: {t | u<t>}; separator=", ", wrap, anchor> ]

    ruleNames = [ <lexer.ruleNames: {r | u"<r>"; separator=", ", wrap, anchor> ]

    grammarFileName = u"<lexer.grammarFileName>"

    def __init__(self, input=None, output=sys.stdout):
        super(<lexer.name>, self).__init__(input, output=output)
        self.checkVersion("<lexerFile.ANTLRVersion>")

```

```

self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
self._actions = None
self._predicates = None

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

def serializedATN():
    with StringIO() as buf:
        buf.write(u"<model.serialized; wrap={ }<\n>      buf.write(u"}>")
        return buf.getvalue()

>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<pythonTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".py"

Found in path(s):
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
jar/org/antlr/v4/tool/templates/codegen/Python2/Python2.stg
No license file was found, but licenses were detected in source scan.

/*
 * [The "BSD license"]
 * Copyright (c) 2012-2016 Terence Parr
 * Copyright (c) 2012-2016 Sam Harwell
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the

```

```

* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** How to generate rules derived from left-recursive rules.
* These rely on recRuleAltPredicate(),
* recRuleArg(), recRuleSetResultAction(), recRuleSetReturnAction()
* templates in main language.stg
*/

```

```
group LeftRecursiveRules;
```

```
recRule(ruleName, argName, primaryAlts, opAlts, setResultAction,
        userRetvals, leftRecursiveRuleRefLabels) ::=
```

```

<<
<ruleName><if(userRetvals)> returns [<userRetvals>]<endif>
: ( { } <primaryAlts:{ alt | <alt.altText> }; separator="\n    | ">
  )
  (
    <opAlts; separator="\n    | ">
  )*
;
>>

```

```

recRuleAlt(alt, precOption, opPrec, pred) ::= <<
{<pred>}?<<precOption>=<opPrec>> <alt.altText>
>>

```

Found in path(s):

```

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
jar/org/antlr/v4/tool/templates/LeftRecursiveRules.stg

```

No license file was found, but licenses were detected in source scan.

```

/*
* [The "BSD license"]
* Copyright (c) 2013 Terence Parr
* Copyright (c) 2013 Sam Harwell

```

* All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

// args must be <object-model-object>, <fields-resulting-in-STs>

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<namedActions.header>
using System;
using System.IO;
using System.Text;
using System.Diagnostics;
using System.Collections.Generic;
using Antlr4.Runtime;
using Antlr4.Runtime.Atn;
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using DFA = Antlr4.Runtime.Dfa.DFA;

<parser>
<if(file.genPackage)>
} // namespace <file.genPackage>
```

```

<endif>
>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using IParseTreeListener = Antlr4.Runtime.Tree.IParseTreeListener;
using IToken = Antlr4.Runtime.IToken;

/// \<summary>
/// This interface defines a complete listener for a parse tree produced by
/// \<see cref="<csIdentifier.(file.parserName)>"/>.
/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Listener : IParseTreeListener {
  <file.listenerNames> { Iname |
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Enter a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>"/>.
<else>
/// Enter a parse tree produced by \<see cref="<file.parserName>.<Iname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Enter<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context
context);
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Exit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>"/>.
<else>
/// Exit a parse tree produced by \<see cref="<file.parserName>.<Iname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Exit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context
context);}; separator="\n"
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

```

```

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>

using Antlr4.Runtime.Misc;
using IErrorNode = Antlr4.Runtime.Tree.IErrorNode;
using ITerminalNode = Antlr4.Runtime.Tree.ITerminalNode;
using IToken = Antlr4.Runtime.IToken;
using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Listener"/>,
/// which can be extended to create a listener which only needs to handle a subset
/// of the available methods.
/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseListener : I<file.grammarName>Listener {
    <file.listenerNames> {Iname |
    /// \<summary>
    <if(file.listenerLabelRuleNames.(Iname))>
    /// Enter a parse tree produced by the \<c><Iname>\</c>
    /// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>">.
    <else>
    /// Enter a parse tree produced by \<see cref="<file.parserName>.<Iname>">.
    <endif>
    /// \<para>The default implementation does nothing.\</para>
    /// \</summary>
    /// \<param name="context">The parse tree.\</param>
    public virtual void Enter<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
    format="cap">Context context) { \}
    /// \<summary>
    <if(file.listenerLabelRuleNames.(Iname))>
    /// Exit a parse tree produced by the \<c><Iname>\</c>
    /// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>">.
    <else>
    /// Exit a parse tree produced by \<see cref="<file.parserName>.<Iname>">.
    <endif>
    /// \<para>The default implementation does nothing.\</para>
    /// \</summary>
    /// \<param name="context">The parse tree.\</param>
    public virtual void Exit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
    format="cap">Context context) { \} }; separator="\n">

```

```

/// \<inheritdoc/>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void EnterEveryRule([NotNull] ParserRuleContext context) { }
/// \<inheritdoc/>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void ExitEveryRule([NotNull] ParserRuleContext context) { }
/// \<inheritdoc/>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void VisitTerminal([NotNull] ITerminalNode node) { }
/// \<inheritdoc/>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void VisitErrorNode([NotNull] IErrorNode node) { }
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

```

```

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;

/// \<summary>
/// This interface defines a complete generic visitor for a parse tree produced
/// by \<see cref="<csIdentifier.(file.parserName)>">/>.
/// \</summary>
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Visitor<Result> : IParseTreeVisitor<Result> {
<file.visitorNames>: { Iname |
/// \<summary>
<if(file.visitorLabelRuleNames.(Iname))>
/// Visit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(Iname)>">/>.
<else>
/// Visit a parse tree produced by \<see cref="<file.parserName>.<Iname>">/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
/// \<return>The visitor result.\</return>
Result Visit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context

```



```

context)); separator="\n">
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;
using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Visitor{Result}" />,
/// which can be extended to create a visitor which only needs to handle a subset
/// of the available methods.
/// \</summary>
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseVisitor<Result> : AbstractParseTreeVisitor<Result>,
I<file.grammarName>Visitor<Result> {
    <file.visitorNames>: { Iname |
/// \<summary>
<if(file.visitorLabelRuleNames.(Iname))>
/// Visit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(Iname)" />.
<else>
/// Visit a parse tree produced by \<see cref="<file.parserName>.<Iname" />.
<endif>
/// \<para>
/// The default implementation returns the result of calling \<see
cref="AbstractParseTreeVisitor{Result}.VisitChildren(IRuleNode)" />
/// on \<paramref name="context" />.
/// \</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
/// \<return>The visitor result.\</return>
public virtual Result Visit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
format="cap">Context context) { return VisitChildren(context); \} }; separator="\n">
}
</if(file.genPackage)>

```

```

} // namespace <file.genPackage>
<endif>
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
//-----
//\<auto-generated>
// This code was generated by a tool.
// ANTLR Version: <ANTLRVersion>
//
// Changes to this file may cause incorrect behavior and will be lost if
// the code is regenerated.
//\</auto-generated>
//-----

// Generated from <grammarFileName> by ANTLR <ANTLRVersion>

// Unreachable code detected
#pragma warning disable 0162
// The variable '...' is assigned but its value is never used
#pragma warning disable 0219
// Missing XML comment for publicly visible type or member '...'
#pragma warning disable 1591
// Ambiguous reference in cref attribute
#pragma warning disable 419

>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <csIdentifier.(parser.name)> : <superClass; null="Parser"> {
protected static DFA[] decisionToDFA;
protected static PredictionContextCache sharedContextCache = new PredictionContextCache();
<if(parser.tokens)>
public const int
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
<endif>
public const int
<parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
public static readonly string[] ruleNames = {
<parser.ruleNames:{r | "<r">}; separator=", ", wrap, anchor>
};
}

```

```

<vocabulary(parser.literalNames, parser.symbolicNames)>

public override string GrammarFileName { get { return "<parser.grammarFileName>"; } }

public override string[] RuleNames { get { return ruleNames; } }

public override string SerializedAtn { get { return new string(_serializedATN); } }

static <csIdentifier.(parser.name)>() {
    decisionToDFA = new DFA[_ATN.NumberOfDecisions];
    for (int i = 0; i < _ATN.NumberOfDecisions; i++) {
        decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
    }
}

<namedActions.members>
<parser:(ctor)()>
<funcs; separator="\n">

<if(sempredFuncs)>
public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <parser.sempredFuncs.values:{f}
case <f.ruleIndex>: return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
    }
    return true;
}
<sempredFuncs.values; separator="\n">
<endif>

<atn>
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
private static readonly string[] _LiteralNames = {
    <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
};
private static readonly string[] _SymbolicNames = {
    <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
};
public static readonly IVocabulary DefaultVocabulary = new Vocabulary(_LiteralNames, _SymbolicNames);

[NotNull]
public override IVocabulary Vocabulary
{
    get
    {

```

```

    return DefaultVocabulary;
}
}
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
public override void Action(RuleContext _localctx, int ruleIndex, int actionIndex) {
    switch (ruleIndex) {
        <recog.actionFuncs.values: {f}
    case <f.ruleIndex> : <f.name>_action(<if(!recog.modes)><f.ctxType><endif>_localctx, actionIndex); break;};
    separator="\n">
    }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <recog.sempredFuncs.values: {f}
    case <f.ruleIndex> : return <f.name>_sempred(<if(!recog.modes)><f.ctxType><endif>_localctx, predIndex);};
    separator="\n">
    }
    return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(parser) ::= <<
public <csIdentifier.(parser.name)>(ITokenStream input) : this(input, Console.Out, Console.Error) { }

public <csIdentifier.(parser.name)>(ITokenStream input, TextWriter output, TextWriter errorOutput)
: base(input, output, errorOutput)
{
    Interpreter = new ParserATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
}
>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
    switch (actionIndex) {
        <actions: {index|
    case <index>: <actions.(index)> break;}; separator="\n">
    }
}

```

```

}
>>

/* This generates a private method since the predIndex is generated, making an
 * overriding implementation impossible to maintain.
 */
RuleSempredFunction(r, actions) ::= <<
private bool <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
  switch (predIndex) {
    <actions:{index|
case <index>: return <actions.(index)>;}; separator="\n">
  }
  return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<if(ruleCtx)>
<ruleCtx>
<endif>
<altLabelCtxs:{l | <altLabelCtxs.(l)>;}; separator="\n">

[RuleVersion(<namedActions.version; null="0">)]
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
<csIdentifier(currentRule.name)><(<args; separator=", ">) {
  <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier(a.name)>>});
  EnterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
  <namedActions.init>
  <locals; separator="\n">
  try {
    <if(currentRule.hasLookaheadBlock)>
    int _alt;
  <endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  catch (RecognitionException re) {
    _localctx.exception = re;
    ErrorHandler.ReportError(this, re);
    ErrorHandler.Recover(this, re);
  }
}

```

```

<endif>
finally {
  <finallyAction>
  ExitRule();
}
return _localctx;
}
>>

```

```

LeftFactoredRuleFunction(currentRule,args,code,locals,namedActions,finallyAction,postamble) ::=
<<

```

```

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>private <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)><(args; separator=", "> {
  <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier.(a.name)>>>);
  EnterLeftFactoredRule(_localctx, <currentRule.startState>, RULE_<currentRule.variantOf>);
  <namedActions.init>
  <locals; separator="\n">
  try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
  catch (RecognitionException re) {
    _localctx.exception = re;
    ErrorHandler.ReportError(this, re);
    ErrorHandler.Recover(this, re);
  }
  finally {
    <finallyAction>
    ExitRule();
  }
  return _localctx;
}
>>

```

```

// This behaves similar to RuleFunction (enterRule is called, and no adjustments
// are made to the parse tree), but since it's still a variant no context class
// needs to be generated.

```

```

LeftUnfactoredRuleFunction(currentRule,args,code,locals,namedActions,finallyAction,postamble) ::=
<<

```

```

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>private <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)><(args; separator=", "> {

```

```

<currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
<csIdentifier.(a.name)>>});
EnterRule(_localctx, <currentRule.startState>, RULE_<currentRule.variantOf>);
<namedActions.init>
<locals; separator="\n">
try {
<if(currentRule.hasLookaheadBlock)>
int _alt;
<endif>
<code>
<postamble; separator="\n">
<namedActions.after>
}
catch (RecognitionException re) {
_localctx.exception = re;
ErrorHandler.ReportError(this, re);
ErrorHandler.Recover(this, re);
}
finally {
<finallyAction>
ExitRule();
}
return _localctx;
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
namedActions,finallyAction,postamble) ::=
<<

```

```

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

[RuleVersion(<namedActions.version; null="0">)]
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
return <csIdentifier.(currentRule.name)>(0<currentRule.args:{a | , <csIdentifier.(a.name)>>});
}

```

```

private <currentRule.ctxType> <csIdentifier.(currentRule.name)>(int _p<args:{a | , <a>}>) {
ParserRuleContext _parentctx = Context;
int _parentState = State;
<currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, _parentState<currentRule.args:{a | ,
<csIdentifier.(a.name)>>});
<currentRule.ctxType> _prevctx = _localctx;
int _startState = <currentRule.startState>;
EnterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
<namedActions.init>

```

```

<locals; separator="\n">
try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
catch (RecognitionException re) {
  _localctx.exception = re;
  ErrorHandler.ReportError(this, re);
  ErrorHandler.Recover(this, re);
  }
finally {
  <finallyAction>
  UnrollRecursionContexts(_parentctx);
  }
return _localctx;
}
>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
EnterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>
<preamble; separator="\n">
switch (TokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break;}; separator="\n">
default:
  <error>

```



```
}  
>>
```

```
LL1OptionalBlock(choice, alts, error) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
switch (TokenStream.LA(1)) {  
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>  
<alt>  
break;}; separator="\n">  
default:  
break;  
}  
>>
```

```
LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
<preamble; separator="\n">  
if (<expr>) {  
<alts; separator="\n">  
}  
<!else if ( !(<followExpr> ) <error!>  
>>
```

```
LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
<preamble; separator="\n">  
while (<loopExpr>) {  
<alts; separator="\n">  
State = <choice.loopBackStateNumber>;  
ErrorHandler.Sync(this);  
<iteration>  
}  
>>
```

```
LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<  
State = <choice.blockStartStateNumber>;<! alt block decision !>  
ErrorHandler.Sync(this);  
<preamble; separator="\n">  
do {  
<alts; separator="\n">  
State = <choice.stateNumber>;<! loopback/exit decision !>  
ErrorHandler.Sync(this);  
<iteration>  
} while ( <loopExpr> );  
>>
```

```
// LL(*) stuff
```

```
AltBlock(choice, preamble, alts, error) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>  
<preamble; separator="\n">  
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {  
<alts:{alt |  
case <i>:  
<alt>  
break;}; separator="\n">  
}  
>>
```

```
OptionalBlock(choice, alts, error) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {  
<alts:{alt |  
case <i><if(!choice.ast.greedy)>+1<endif>:  
<alt>  
break;}; separator="\n">  
}  
>>
```

```
StarBlock(choice, alts, sync, iteration) ::= <<  
State = <choice.stateNumber>;  
ErrorHandler.Sync(this);  
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);  
while ( !_alt!=<choice.exitAlt> && !_alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER ) {  
if ( !_alt==1<if(!choice.ast.greedy)>+1<endif> ) {  
<iteration>  
<alts> <! should only be one !>  
}  
State = <choice.loopBackStateNumber>;  
ErrorHandler.Sync(this);  
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);  
}  
>>
```

```
PlusBlock(choice, alts, error) ::= <<  
State = <choice.blockStartStateNumber>;<! alt block decision !>  
ErrorHandler.Sync(this);  
_alt = 1<if(!choice.ast.greedy)>+1<endif>;  
do {  
switch ( _alt ) {
```

```

<alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
default:
<error>
}
State = <choice.loopBackStateNumber>;<! loopback/exit decision !>
ErrorHandler.Sync(this);
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
} while ( _alt!=<choice.exitAlt> && _alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "Sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ ttype | (1L \<< <offsetShift(tokenType.(ttype), bits.shift)>)}; separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ ttype | <s.varName>==<tokenType.(ttype)>}; separator=" || ">
%>

cases(ttypes) ::= <<

```

```

<ttypes:{t | case <tokenType.(t)>}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
State = <r.stateNumber>; <if(r.labels)><r.labels:{1 | <labelref(l)> =
}><endif><csIdentifier.(r.name)><(if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)><endif><endif><arg
ExprsChunks>;
>>

MatchToken(m) ::= <<
State = <m.stateNumber>; <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>Match(<tokenType.(m.name)>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
State = <m.stateNumber>;
<if(m.labels)><m.labels:{1 | <labelref(l)> = }>TokenStream.LT(1);<endif>
<capture>
if ( <if(invert)><m.varName> \<= 0 || <else>!<endif><(expr)> ) {
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>ErrorHandler.RecoverInline(this);
}
else {
  ErrorHandler.ReportMatch(this);
  Consume();
}
>>

Wildcard(w) ::= <<
State = <w.stateNumber>;
<if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>MatchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
State = <p.stateNumber>;
if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
>>

ExceptionClause(e, catchArg, catchAction) ::= <<

```

```

catch (<catchArg>) {
  <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "Skip();"
LexerMoreCommand() ::= "More();"
LexerPopModeCommand() ::= "PopMode();"

LexerTypeCommand(arg, grammar) ::= "_type = <tokenType.(arg)>";
LexerChannelCommand(arg, grammar) ::= "_channel = <channelName.(arg)>";
LexerModeCommand(arg, grammar) ::= "_mode = <modeName.(arg)>";
LexerPushModeCommand(arg, grammar) ::= "PushMode(<modeName.(arg)>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
LocalRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
RetValRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<csIdentifier.(a.name)>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<csIdentifier.(tokenType.(t.name))>"
LabelRef(t) ::= "<ctx(t)>.<csIdentifier.(t.name)>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(csIdentifier.(t.name))>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<csIdentifier.(s.name)> = <rhsChunks>";

TokenLabelType() ::= "<file.TokenLabelType; null={IToken}>"
InputSymbolType() ::= "<file.InputSymbolType; null={IToken}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Text:null)"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Type:0)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Line:0)"
TokenPropertyRef_pos(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.CharPositionInLine:0)"
TokenPropertyRef_channel(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Channel:0)"
TokenPropertyRef_index(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.TokenIndex:0)"
TokenPropertyRef_int(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?int.Parse(<ctx(t)>.<tokenType.(t.label)>.Text):0)"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Start):null)"
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_text(r) ::=
"<ctx(r)>.<r.label>!=null?TokenStream.GetText(<ctx(r)>.<r.label>.Start,<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"

```

```
RulePropertyRef_parser(r) ::= "this"
```

```
ThisRulePropertyRef_start(r) ::= "_localctx.Start"
```

```
ThisRulePropertyRef_stop(r) ::= "_localctx.Stop"
```

```
ThisRulePropertyRef_text(r) ::= "TokenStream.GetText(_localctx.Start, TokenStream.LT(-1))"
```

```
ThisRulePropertyRef_ctx(r) ::= "_localctx"
```

```
ThisRulePropertyRef_parser(r) ::= "this"
```

```
NonLocalAttrRef(s) ::= <%((<s.ruleName;
```

```
format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)>%>
```

```
SetNonLocalAttr(s, rhsChunks) ::=
```

```
<%((<s.ruleName; format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)> =  
<rhsChunks>;%>
```

```
AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.Add(<labelref(a.label)>);"
```

```
TokenDecl(t) ::= "<TokenLabelType() <csIdentifier.(tokenType.(t.name))>"
```

```
TokenTypeDecl(t) ::= "int <csIdentifier.(tokenType.(t.name))>";"
```

```
TokenListDecl(t) ::= "IList<IToken> <csIdentifier.(tokenType.(t.name))> = new List<IToken>()"
```

```
RuleContextDecl(r) ::= "<r.ctxName> <csIdentifier.(r.name)>"
```

```
RuleContextListDecl(rdecl) ::= "IList<<rdecl.ctxName>> <csIdentifier.(rdecl.name)> = new  
List<<rdecl.ctxName>>()"
```

```
contextGetterCollection(elementType) ::= <%
```

```
<elementType>[]
```

```
%>
```

```
ContextTokenGetterDecl(t) ::=
```

```
"public ITerminalNode <csIdentifier.(tokenType.(t.name))>() { return
```

```
GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, 0); }"
```

```
ContextTokenListGetterDecl(t) ::= <<
```

```
public <contextGetterCollection("ITerminalNode")> <csIdentifier.(tokenType.(t.name))>() { return  
GetTokens(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>); }
```

```
>>
```

```
ContextTokenListIndexedGetterDecl(t) ::= <<
```

```
public ITerminalNode <csIdentifier.(tokenType.(t.name))>(int i) {
```

```
return GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, i);
```

```
}
```

```
>>
```

```
ContextRuleGetterDecl(r) ::= <<
```

```
public <r.ctxName> <csIdentifier.(r.name)>() {
```

```
return GetRuleContext<<r.ctxName>>()>();
```

```
}
```

```
>>
```

```
ContextRuleListGetterDecl(r) ::= <<
```

```
public <contextGetterCollection({<r.ctxName>})> <csIdentifier.(r.name)>() {
```

```
return GetRuleContexts<<r.ctxName>>()>();
```

```
}
```

```

>>
ContextRuleListIndexedGetterDecl(r) ::= <<
public <r.ctxName> <csIdentifier.(r.name)>(int i) {
    return GetRuleContext\<<r.ctxName>\>(i);
}
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenType.(tokenName)>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "_<label>"

CaptureNextToken(d) ::= "<d.varName> = TokenStream.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = TokenStream.LA(1);"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers,
    superClass={ParserRuleContext}) ::= <<
public partial class <struct.name> :
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces;
separator=", "><endif> {
    <attrs:{a | public <a>;}; separator="\n">
    <getters:{g | <g>;}; separator="\n">
    <if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) : base(parent, invokingState) {
    }<endif>
    public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{a | , <a>>)>
    : base(parent, invokingState)
    {
        <struct.ctorAttrs:{a | this.<csIdentifier.(a.name)> = <csIdentifier.(a.name)>;}; separator="\n">
    }
    public override int RuleIndex { get { return RULE_<struct.derivedFromName>; } }
    <if(struct.provideCopyFrom)><! don't need copy unless we have subclasses !>
    public <struct.name>() { }
    public virtual void CopyFrom(<struct.name> context) {
        base.CopyFrom(context);
        <struct.attrs:{a | this.<csIdentifier.(a.name)> = context.<csIdentifier.(a.name)>;}; separator="\n">
    }
    <endif>
    <dispatchMethods; separator="\n">
    <extensionMembers; separator="\n">
}
>>

```

```

AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
public partial class <struct.name> : <currentRule.name; format="cap">Context {
  <attrs:{ a | public <a>; }; separator="\n">
  <getters:{ g | <g>; }; separator="\n">
  public <struct.name>(<currentRule.name; format="cap">Context context) { CopyFrom(context); }
  <dispatchMethods; separator="\n">
}
>>

```

```

ListenerDispatchMethod(method) ::= <<
public override void <if(method.isEnter)>Enter<else>Exit<endif>Rule(IParseTreeListener listener) {
  I<parser.grammarName>Listener typedListener = listener as I<parser.grammarName>Listener;
  if (typedListener != null) typedListener.<if(method.isEnter)>Enter<else>Exit<endif><struct.derivedFromName;
  format="cap">(this);
}
>>

```

```

VisitorDispatchMethod(method) ::= <<
public override TResult Accept<TResult>(IParseTreeVisitor<TResult> visitor) {
  I<parser.grammarName>Visitor<TResult> typedVisitor = visitor as I<parser.grammarName>Visitor<TResult>;
  if (typedVisitor != null) return typedVisitor.Visit<struct.derivedFromName; format="cap">(this);
  else return visitor.VisitChildren(this);
}
>>

```

```

AttributeDecl(d) ::= "<d.type> <csIdentifier.(d.name)><if(d.initValue)> = <d.initValue><endif>"

```

```

/** If we don't know location of label def x, use this template */

```

```

labelref(x) ::= "<if(!x.isLocal)><typedContext(x.ctx)>.<endif><csIdentifier.(x.name)>"

```

```

/** For any action chunk, what is correctly-typed context struct ptr? */

```

```

ctx(actionChunk) ::= "<typedContext(actionChunk.ctx)>"

```

```

// only casts _localctx to the type when the cast isn't redundant (i.e. to a sub-context for a labeled alt)

```

```

typedContext(ctx) ::= "<if(ctx.provideCopyFrom)>((<ctx.name>)_localctx)<else>_localctx<endif>"

```

```

// used for left-recursive rules

```

```

recRuleAltPredicate(ruleName,opPrec) ::= "Precpred(Context, <opPrec>)"

```

```

recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>;"

```

```

recRuleSetStopToken() ::= "Context.Stop = TokenStream.LT(-1);"

```

```

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<

```

```

  _localctx = new <ctxName>Context(_parentctx, _parentState);

```

```

  <if(label)>

```

```

  <if(isListLabel)>

```

```

  _localctx.<label>.Add(_prevctx);

```

```

  <else>

```



```

_localctx.<label> = _prevctx;
<endif>
<endif>
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
<if(label)>
<if(isListLabel)>
((<currentAltLabel; format="cap">Context)_localctx).<label>.Add(_prevctx);
<else>
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;
<endif>
<endif>
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = new <ctxName>Context(_localctx);
Context = _localctx;
_prevctx = _localctx;
>>

recRuleSetPrevCtx() ::= <<
if ( ParseListeners!=null )
    TriggerExitRuleEvent();
_prevctx = _localctx;
>>

LexerFile(file, lexer, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<namedActions.header>
using System;
using System.IO;
using System.Text;
using Antlr4.Runtime;
using Antlr4.Runtime.Atn;
using Antlr4.Runtime.Misc;
using DFA = Antlr4.Runtime.Dfa.DFA;

<lexer>
<if(file.genPackage)>

```

```

} // namespace <file.genPackage>
<endif>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <csIdentifier.(lexer.name)> : <superClass; null="Lexer"> {
protected static DFA[] decisionToDFA;
protected static PredictionContextCache sharedContextCache = new PredictionContextCache();
public const int
<lexer.tokens:{k | <tokenType.(k)>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>;
<if(lexer.channels)>
public const int
<lexer.channels:{k | <csIdentifier.(k)>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>;
<endif>
<if(rest(lexer.modes))>
public const int
<rest(lexer.modes){m | <m>=<i>}; separator=", ", wrap, anchor>;
<endif>
public static string[] channelNames = {
"DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c | "<c>"}; separator=", ",
wrap, anchor><endif>
};

public static string[] modeNames = {
<lexer.modes:{m | "<m>"}; separator=", ", wrap, anchor>
};

public static readonly string[] ruleNames = {
<lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};

<namedActions.members>

public <csIdentifier.(lexer.name)>(ICharStream input)
: this(input, Console.Out, Console.Error) { }

public <csIdentifier.(lexer.name)>(ICharStream input, TextWriter output, TextWriter errorOutput)
: base(input, output, errorOutput)
{
Interpreter = new LexerATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
}

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

public override string GrammarFileName { get { return "<lexer.grammarFileName>"; } }

```

```

public override string[] RuleNames { get { return ruleNames; } }

public override string[] ChannelNames { get { return channelNames; } }

public override string[] ModeNames { get { return modeNames; } }

public override string SerializedAtn { get { return new string(_serializedATN); } }

static <csIdentifier.(lexer.name)>() {
    decisionToDFA = new DFA[_ATN.NumberOfDecisions];
    for (int i = 0; i < _ATN.NumberOfDecisions; i++) {
        decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
    }
}
<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
}
>>

```

```

SerializedATN(model) ::= <<
private static char[] _serializedATN = {
    <model.serialized; separator=", ", wrap>,
};

public static readonly ATN _ATN =
    new ATNDeserializer().Deserialize(_serializedATN);

```

```
>>
```

```

initValue(typeName) ::= <<
default(<typeName>)
>>

```

```
codeFileExtension() ::= ".cs"
```

```

modeName ::= [
    "DEFAULT_MODE" : "DefaultMode",
    default : key
]

```

```

channelName ::= [
    "HIDDEN" : "Hidden",
    "DEFAULT_TOKEN_CHANNEL" : "DefaultTokenChannel",
    default : key
]

```

```
tokenType ::= [  
  "EOF" : "Eof",  
  default : key  
]
```

```
csIdentifier ::= [  
  "abstract" : "@abstract",  
  "as" : "@as",  
  "base" : "@base",  
  "bool" : "@bool",  
  "break" : "@break",  
  "byte" : "@byte",  
  "case" : "@case",  
  "catch" : "@catch",  
  "char" : "@char",  
  "checked" : "@checked",  
  "class" : "@class",  
  "const" : "@const",  
  "continue" : "@continue",  
  "decimal" : "@decimal",  
  "default" : "@default",  
  "delegate" : "@delegate",  
  "do" : "@do",  
  "double" : "@double",  
  "else" : "@else",  
  "enum" : "@enum",  
  "event" : "@event",  
  "explicit" : "@explicit",  
  "extern" : "@extern",  
  "false" : "@false",  
  "finally" : "@finally",  
  "fixed" : "@fixed",  
  "float" : "@float",  
  "for" : "@for",  
  "foreach" : "@foreach",  
  "goto" : "@goto",  
  "if" : "@if",  
  "implicit" : "@implicit",  
  "in" : "@in",  
  "int" : "@int",  
  "interface" : "@interface",  
  "internal" : "@internal",  
  "is" : "@is",  
  "lock" : "@lock",  
  "long" : "@long",  
  "namespace" : "@namespace",  
  "new" : "@new",  
  "null" : "@null",
```

```
"object" : "@object",
"operator" : "@operator",
"out" : "@out",
"override" : "@override",
"params" : "@params",
"private" : "@private",
"protected" : "@protected",
"public" : "@public",
"readonly" : "@readonly",
"ref" : "@ref",
"return" : "@return",
"sbyte" : "@sbyte",
"sealed" : "@sealed",
"short" : "@short",
"sizeof" : "@sizeof",
"stackalloc" : "@stackalloc",
"static" : "@static",
"string" : "@string",
"struct" : "@struct",
"switch" : "@switch",
"this" : "@this",
"throw" : "@throw",
>true" : "@true",
"try" : "@try",
"typeof" : "@typeof",
"uint" : "@uint",
"ulong" : "@ulong",
"unchecked" : "@unchecked",
"unsafe" : "@unsafe",
"ushort" : "@ushort",
"using" : "@using",
"virtual" : "@virtual",
"values" : "@values",
"void" : "@void",
"volatile" : "@volatile",
"while" : "@while",
default : key
]
```

Found in path(s):

```
* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-
jar/org/antlr/v4/tool/templates/codegen/CSharp/CSharp.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2016, Mike Lischke
```

```
* All rights reserved.
```

```

*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

Found in path(s):

```

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/codegen/Cpp/Files.stg
No license file was found, but licenses were detected in source scan.

```

```

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

```

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

```
javaTypeInitMap ::= [
  "int": "0",
  "long": "0",
  "float": "0.0f",
  "double": "0.0",
  "boolean": "false",
  "byte": "0",
  "short": "0",
  "char": "0",
  default: "null" // anything other than a primitive type is an object
]
```

// args must be <object-model-object>, <fields-resulting-in-STs>

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<namedActions.header>
import org.antlr.v4.runtime.atn.*;
import org.antlr.v4.runtime.dfa.DFA;
import org.antlr.v4.runtime.*;
import org.antlr.v4.runtime.misc.*;
import org.antlr.v4.runtime.tree.*;
import java.util.List;
import java.util.Iterator;
import java.util.ArrayList;

<parser>
>>
```

```
ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
```

```

<header>
import org.antlr.v4.runtime.tree.ParseTreeListener;

/**
 * This interface defines a complete listener for a parse tree produced by
 * { @link <file.parserName> }.
 */
public interface <file.grammarName>Listener extends ParseTreeListener {
    <file.listenerNames: {Iname |
/**
<if(file.listenerLabelRuleNames.(Iname))>
 * Enter a parse tree produced by the { @code <Iname>\}
 * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>\}.
<else>
 * Enter a parse tree produced by { @link <file.parserName>#<Iname>\}.
<endif>
 * @param ctx the parse tree
 */
void enter<Iname; format="cap">(<file.parserName>.<Iname; format="cap">Context ctx);
/**
<if(file.listenerLabelRuleNames.(Iname))>
 * Exit a parse tree produced by the { @code <Iname>\}
 * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>\}.
<else>
 * Exit a parse tree produced by { @link <file.parserName>#<Iname>\}.
<endif>
 * @param ctx the parse tree
 */
void exit<Iname; format="cap">(<file.parserName>.<Iname; format="cap">Context ctx);}; separator="\n">
}
>>

```

```

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>

```

```

import org.antlr.v4.runtime.ParserRuleContext;
import org.antlr.v4.runtime.tree.ErrorNode;
import org.antlr.v4.runtime.tree.TerminalNode;

```

```

/**
 * This class provides an empty implementation of { @link <file.grammarName>Listener },
 * which can be extended to create a listener which only needs to handle a subset
 * of the available methods.
 */

```



```

public class <file.grammarName>BaseListener implements <file.grammarName>Listener {
    <file.listenerNames: {lname |
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void enter<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { \}
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void exit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { \};
separator="\n">

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void enterEveryRule(ParserRuleContext ctx) { }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void exitEveryRule(ParserRuleContext ctx) { }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitTerminal(TerminalNode node) { }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitErrorNode(ErrorNode node) { }
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>

```

```

<header>
import org.antlr.v4.runtime.tree.ParseTreeVisitor;

/**
 * This interface defines a complete generic visitor for a parse tree produced
 * by { @link <file.parserName> }.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void } for
 * operations with no return type.
 */
public interface <file.grammarName>Visitor\<T> extends ParseTreeVisitor\<T> {
    <file.visitorNames:{ lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
 * Visit a parse tree produced by the { @code <lname>\}
 * labeled alternative in { @link <file.parserName>#<file.visitorLabelRuleNames.(lname)>\}.
<else>
 * Visit a parse tree produced by { @link <file.parserName>#<lname>\}.
<endif>
 * @param ctx the parse tree
 * @return the visitor result
 */
T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx);}; separator="\n"
}
>>

```

```

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.AbstractParseTreeVisitor;

/**
 * This class provides an empty implementation of { @link <file.grammarName>Visitor },
 * which can be extended to create a visitor which only needs to handle a subset
 * of the available methods.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void } for
 * operations with no return type.
 */
public class <file.grammarName>BaseVisitor\<T> extends AbstractParseTreeVisitor\<T> implements
<file.grammarName>Visitor\<T> {
    <file.visitorNames:{ lname |
/**
 * { @inheritDoc\}
 *

```

```

* \<p>The default implementation returns the result of calling
* { @link #visitChildren\} on { @code ctx\}.\</p>
*/
@Override public T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { return
visitChildren(ctx); \} }; separator="\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <parser.name> extends <superClass; null="Parser"> {
static { RuntimeMetaData.checkVersion("<file.ANTLRVersion>", RuntimeMetaData.VERSION); }

protected static final DFA[] _decisionToDFA;
protected static final PredictionContextCache _sharedContextCache =
new PredictionContextCache();
<if(parser.tokens)>
public static final int
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
<endif>
public static final int
<parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
private static String[] makeRuleNames() {
return new String[] {
<parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};
}
public static final String[] ruleNames = makeRuleNames();

<vocabulary(parser.literalNames, parser.symbolicNames)>

@Override
public String getGrammarFileName() { return "<parser.grammarFileName; format="java-escape">"; }

@Override
public String[] getRuleNames() { return ruleNames; }

@Override
public String getSerializedATN() { return _serializedATN; }

```

```

@Override
public ATN getATN() { return _ATN; }

<namedActions.members>
<parser:(ctor)()>
<funcs; separator="\n">

<if(sempredFuncs)>
public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <parser.sempredFuncs.values:{f}
case <f.ruleIndex>:
return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
    }
    return true;
    }
    <sempredFuncs.values; separator="\n">
<endif>

<atn>
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
private static String[] makeLiteralNames() {
    return new String[] {
        <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
    };
}
private static final String[] _LITERAL_NAMES = makeLiteralNames();
private static String[] makeSymbolicNames() {
    return new String[] {
        <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
    };
}
private static final String[] _SYMBOLIC_NAMES = makeSymbolicNames();
public static final Vocabulary VOCABULARY = new VocabularyImpl(_LITERAL_NAMES,
    _SYMBOLIC_NAMES);

/**
 * @deprecated Use {@link #VOCABULARY} instead.
 */
@Deprecated
public static final String[] tokenNames;
static {
    tokenNames = new String[_SYMBOLIC_NAMES.length];
    for (int i = 0; i < tokenNames.length; i++) {
        tokenNames[i] = VOCABULARY.getLiteralName(i);
    }
}

```

```

if (tokenNames[i] == null) {
    tokenNames[i] = VOCABULARY.getSymbolicName(i);
}

```

```

if (tokenNames[i] == null) {
    tokenNames[i] = "\\<INVALID>";
}
}
}

```

```

@Override
@Deprecated
public String[] getTokenNames() {
    return tokenNames;
}

```

```

@Override

```

```

public Vocabulary getVocabulary() {
    return VOCABULARY;
}
>>

```

```

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
@Override
public void action(RuleContext _localctx, int ruleIndex, int actionIndex) {
    switch (ruleIndex) {
        <recog.actionFuncs.values:{f}
        case <f.ruleIndex>:
            <f.name>_action((<f.ctxType>)_localctx, actionIndex);
            break;}; separator="\n"
    }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
@Override
public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <recog.sempredFuncs.values:{f}
        case <f.ruleIndex>:
            return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n"
    }
    return true;
}
<sempredFuncs.values; separator="\n">
<endif>

```

>>

```
parser_ctor(p) ::= <<
public <p.name>(TokenStream input) {
    super(input);
    _interp = new ParserATNSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
}
>>
```

```
/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
```

```
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
    switch (actionIndex) {
        <actions:{index|
case <index>:
        <actions.(index)>
        break;}; separator="\n">
    }
}
>>
```

```
/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
```

```
RuleSempredFunction(r, actions) ::= <<
private boolean <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
    switch (predIndex) {
        <actions:{index|
case <index>:
        return <actions.(index)>;}; separator="\n">
    }
    return true;
}
>>
```

```
RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<
```

```
<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name>(<args; separator=",">) throws RecognitionException {
    <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, getState())<currentRule.args:{a | ,
<a.name>}>>;
    enterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
```

```

<namedActions.init>
<locals; separator="\n">
try {
<if(currentRule.hasLookaheadBlock)>
    int _alt;
<endif>
    <code>
    <postamble; separator="\n">
    <namedActions.after>
    }
<if(exceptions)>
<exceptions; separator="\n">
<else>
catch (RecognitionException re) {
    _localctx.exception = re;
    _errHandler.reportError(this, re);
    _errHandler.recover(this, re);
}
<endif>
finally {
    <finallyAction>
    exitRule();
}
return _localctx;
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
namedActions,finallyAction,postamble) ::=
<<

```

```

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name><(<args; separator=", ">) throws RecognitionException {
return <currentRule.name>(0<currentRule.args:{a | , <a.name>}>);
}

```

```

private <currentRule.ctxType> <currentRule.name>(int _p<args:{a | , <a>}>) throws RecognitionException {
    ParserRuleContext _parentctx = _ctx;
    int _parentState = getState();
    <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
<a.name>}>);
    <currentRule.ctxType> _prevctx = _localctx;
    int _startState = <currentRule.startState>;
    enterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
    <namedActions.init>

```

```

<locals; separator="\n">
try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
catch (RecognitionException re) {
  _localctx.exception = re;
  _errHandler.reportError(this, re);
  _errHandler.recover(this, re);
}
finally {
  <finallyAction>
  unrollRecursionContexts(_parentctx);
}
return _localctx;
}
>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>
<preamble; separator="\n">
switch (_input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break;}; separator="\n">
default:
  <error>

```



```
}  
>>
```

```
LL1OptionalBlock(choice, alts, error) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  switch (_input.LA(1)) {  
  <choice.altLook,alts:{look,alt| <cases(ttypes=look)>  
  <alt>  
  break;}; separator="\n">  
  default:  
  break;  
  }  
>>
```

```
LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  <preamble; separator="\n">  
  if (<expr>) {  
  <alts; separator="\n">  
  }  
  <!else if ( !(<followExpr>) ) <error!>  
>>
```

```
LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  <preamble; separator="\n">  
  while (<loopExpr>) {  
  <alts; separator="\n">  
  setState(<choice.loopBackStateNumber>);  
  _errHandler.sync(this);  
  <iteration>  
  }  
>>
```

```
LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<  
  setState(<choice.blockStartStateNumber>); <! alt block decision !>  
  _errHandler.sync(this);  
  <preamble; separator="\n">  
  do {  
  <alts; separator="\n">  
  setState(<choice.stateNumber>); <! loopback/exit decision !>  
  _errHandler.sync(this);  
  <iteration>  
  } while ( <loopExpr> );  
>>
```

```
// LL(*) stuff
```

```
AltBlock(choice, preamble, alts, error) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  <if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>  
  <preamble; separator="\n">  
  switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {  
  <alts:{alt |  
  case <i>:  
    <alt>  
    break;}; separator="\n">  
  }  
>>
```

```
OptionalBlock(choice, alts, error) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {  
  <alts:{alt |  
  case <i><if(!choice.ast.greedy)>+1<endif>:  
    <alt>  
    break;}; separator="\n">  
  }  
>>
```

```
StarBlock(choice, alts, sync, iteration) ::= <<  
  setState(<choice.stateNumber>);  
  _errHandler.sync(this);  
  _alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);  
  while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER ) {  
  if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {  
    <iteration>  
    <alts> <! should only be one !>  
  }  
  setState(<choice.loopBackStateNumber>);  
  _errHandler.sync(this);  
  _alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);  
  }  
>>
```

```
PlusBlock(choice, alts, error) ::= <<  
  setState(<choice.blockStartStateNumber>); <! alt block decision !>  
  _errHandler.sync(this);  
  _alt = 1<if(!choice.ast.greedy)>+1<endif>;  
  do {  
    switch (_alt) {
```

```

<alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
default:
<error>
}
setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
_errHandler.sync(this);
_alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
} while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ ttype | (1L \<< <offsetShift(ttype, bits.shift)>)}; separator=" | ">)) != 0
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ ttype | <s.varName>==<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<

```

```
<ttypes:{t | case <t>:}; separator="\n">
```

```
>>
```

```
InvokeRule(r, argExprsChunks) ::= <<
```

```
  setState(<r.stateNumber>);
```

```
  <if(r.labels)><r.labels:{1 | <labelref(l)> =
```

```
>><endif><r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>
```

```
);
```

```
>>
```

```
MatchToken(m) ::= <<
```

```
  setState(<m.stateNumber>);
```

```
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>match(<m.name>);
```

```
>>
```

```
MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"
```

```
MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"
```

```
CommonSetStuff(m, expr, capture, invert) ::= <<
```

```
  setState(<m.stateNumber>);
```

```
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }>>_input.LT(1);<endif>
```

```
  <capture>
```

```
  if ( <if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr> ) ) {
```

```
    <if(m.labels)><m.labels:{1 | <labelref(l)> = (Token)}><endif>_errHandler.recoverInline(this);
```

```
  }
```

```
  else {
```

```
    if ( _input.LA(1)==Token.EOF ) matchedEOF = true;
```

```
    _errHandler.reportMatch(this);
```

```
    consume();
```

```
  }
```

```
>>
```

```
Wildcard(w) ::= <<
```

```
  setState(<w.stateNumber>);
```

```
  <if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>matchWildcard();
```

```
>>
```

```
// ACTION STUFF
```

```
Action(a, foo, chunks) ::= "<chunks>"
```

```
ArgAction(a, chunks) ::= "<chunks>"
```

```
SemPred(p, chunks, failChunks) ::= <<
```

```
  setState(<p.stateNumber>);
```

```
  if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,<
```

```
<failChunks><elseif(p.msg)>, <p.msg><endif>);
```

>>

```
ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
  <catchAction>
}
>>
```

// lexer actions are not associated with model objects

```
LexerSkipCommand() ::= "skip();"
LexerMoreCommand() ::= "more();"
LexerPopModeCommand() ::= "popMode();"

LexerTypeCommand(arg, grammar) ::= "_type = <arg>";
LexerChannelCommand(arg, grammar) ::= "_channel = <arg>";
LexerModeCommand(arg, grammar) ::= "_mode = <arg>";
LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>);"
```

```
ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValRef(a) ::= "_localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>";"
```

```
TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"
```

```
TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getText():null)"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getType():0)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getLine():0)"
TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getCharPositionInLine():0)"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getChannel():0)"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getTokenIndex():0)"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label>!=null?Integer.valueOf(<ctx(t)>.<t.label>.getText():0)"
```

```
RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>!=null?<ctx(r)>.<r.label>.start():null)"
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>!=null?<ctx(r)>.<r.label>.stop():null)"
RulePropertyRef_text(r) ::=
"<ctx(r)>.<r.label>!=null?_input.getText(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop):null)"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "this"
```

```

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r) ::= "_localctx.stop"
ThisRulePropertyRef_text(r) ::= "_input.getText(_localctx.start, _input.LT(-1))"
ThisRulePropertyRef_ctx(r) ::= "_localctx"
ThisRulePropertyRef_parser(r) ::= "this"

NonLocalAttrRef(s) ::= "(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::=
"(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name> = <rhsChunks>);"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.add(<labelref(a.label)>);"

TokenDecl(t) ::= "<TokenLabelType() <t.name>"
TokenTypeDecl(t) ::= "int <t.name>);"
TokenListDecl(t) ::= "List<Token> <t.name> = new ArrayList<Token>()"
RuleContextDecl(r) ::= "<r.ctxName> <r.name>"
RuleContextListDecl(rdecl) ::= "List<<rdecl.ctxName>> <rdecl.name> = new ArrayList<<rdecl.ctxName>>()"

ContextTokenGetterDecl(t) ::=
"public TerminalNode <t.name>() { return getToken(<parser.name>.<t.name>, 0); }"
ContextTokenListGetterDecl(t) ::=
"public List<TerminalNode> <t.name>() { return getTokens(<parser.name>.<t.name>); }"
ContextTokenListIndexedGetterDecl(t) ::= <<
public TerminalNode <t.name>(int i) {
return getToken(<parser.name>.<t.name>, i);
}
>>
ContextRuleGetterDecl(r) ::= <<
public <r.ctxName> <r.name>() {
return getRuleContext(<r.ctxName>.class,0);
}
>>
ContextRuleListGetterDecl(r) ::= <<
public List<<r.ctxName>> <r.name>() {
return getRuleContexts(<r.ctxName>.class);
}
>>
ContextRuleListIndexedGetterDecl(r) ::= <<
public <r.ctxName> <r.name>(int i) {
return getRuleContext(<r.ctxName>.class,i);
}
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.

```

```

*/
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "<tokenName>"
ImplicitRuleLabel(ruleName) ::= "<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = _input.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = _input.LA(1);"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers)
 ::= <<
public static class <struct.name> extends
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)> implements
<interfaces; separator=", "><endif> {
<attrs:{ a | public <a>;}; separator="\n">
<getters:{ g | <g>;}; separator="\n">
<if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) { super(parent, invokingState);
}<endif>
public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{ a | , <a>>}) {
super(parent, invokingState);
<struct.ctorAttrs:{ a | this.<a.name> = <a.name>;}; separator="\n">
}
@Override public int getRuleIndex() { return RULE_<struct.derivedFromName>; }
<if(struct.provideCopyFrom)><! don't need copy unless we have subclasses !>
public <struct.name>() { }
public void copyFrom(<struct.name> ctx) {
super.copyFrom(ctx);
<struct.attrs:{ a | this.<a.name> = ctx.<a.name>;}; separator="\n">
}
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">
}
>>

AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
public static class <struct.name> extends <currentRule.name; format="cap">Context {
<attrs:{ a | public <a>;}; separator="\n">
<getters:{ g | <g>;}; separator="\n">
public <struct.name>(<currentRule.name; format="cap">Context ctx) { copyFrom(ctx); }
<dispatchMethods; separator="\n">
}
>>

ListenerDispatchMethod(method) ::= <<
@Override

```

```

public void <if(method.isEnter)>enter<else>exit<endif>Rule(ParseTreeListener listener) {
    if ( listener instanceof <parser.grammarName>Listener )
        ((<parser.grammarName>Listener)listener).<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName;
        format="cap">(this);
    }
}
>>

```

```

VisitorDispatchMethod(method) ::= <<
@Override
public <T> T accept(ParseTreeVisitor<? extends T> visitor) {
    if ( visitor instanceof <parser.grammarName>Visitor ) return ((<parser.grammarName>Visitor<? extends
    T>)visitor).visit<struct.derivedFromName; format="cap">(this);
    else return visitor.visitChildren(this);
}
}
>>

```

```

AttributeDecl(d) ::= "<d.type> <d.name><if(d.initValue)> = <d.initValue><endif>"

```

```

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>((<x.ctx.name>)_localctx).<endif><x.name>"

```

```

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "((<actionChunk.ctx.name>)_localctx)"

```

```

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "precpred(_ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>;"
recRuleSetStopToken() ::= "_ctx.stop = _input.LT(-1);"

```

```

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
    _localctx = new <ctxName>Context(_parentctx, _parentState);
    <if(label)>
    <if(isListLabel)>
        _localctx.<label>.add(_prevctx);
    <else>
        _localctx.<label> = _prevctx;
    <endif>
    <endif>
    <if(label)>_localctx.<label> = _prevctx;<endif>
    pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
}
>>

```

```

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
    _localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
    _parentState));
    <if(label)>
    <if(isListLabel)>
        ((<currentAltLabel; format="cap">Context)_localctx).<label>.add(_prevctx);

```



```

<else>
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;
<endif>
<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = new <ctxName>Context(_localctx);
_ctx = _localctx;
_prevctx = _localctx;
>>

recRuleSetPrevCtx() ::= <<
if ( _parseListeners!=null ) triggerExitRuleEvent();
_prevctx = _localctx;
>>

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
package <lexerFile.genPackage>;
<endif>
<namedActions.header>
import org.antlr.v4.runtime.Lexer;
import org.antlr.v4.runtime.CharStream;
import org.antlr.v4.runtime.Token;
import org.antlr.v4.runtime.TokenStream;
import org.antlr.v4.runtime.*;
import org.antlr.v4.runtime.atn.*;
import org.antlr.v4.runtime.dfa.DFA;
import org.antlr.v4.runtime.misc.*;

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <lexer.name> extends <superClass; null="Lexer"> {
static { RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION); }

protected static final DFA[] _decisionToDFA;
protected static final PredictionContextCache _sharedContextCache =
new PredictionContextCache();
<if(lexer.tokens)>
public static final int
<lexer.tokens:{k | <k>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>;

```

```

<endif>
<if(lexer.channels)>
public static final int
  <lexer.channels:{c | <c>=<lexer.channels.(c)>}; separator=", ", wrap, anchor>;
<endif>
<if(rest(lexer.modes))>
public static final int
  <rest(lexer.modes){m | <m>=<i>}; separator=", ", wrap, anchor>;
<endif>
public static String[] channelNames = {
  "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
};

public static String[] modeNames = {
  <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
};

private static String[] makeRuleNames() {
  return new String[] {
    <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
  };
}
public static final String[] ruleNames = makeRuleNames();

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

<namedActions.members>

public <lexer.name>(CharStream input) {
  super(input);
  _interp = new LexerATNSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
}

@Override
public String getGrammarFileName() { return "<lexer.grammarFileName>"; }

@Override
public String[] getRuleNames() { return ruleNames; }

@Override
public String getSerializedATN() { return _serializedATN; }

@Override
public String[] getChannelNames() { return channelNames; }

@Override
public String[] getModeNames() { return modeNames; }

```

```

@Override
public ATN getATN() { return _ATN; }

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
}
>>

SerializedATN(model) ::= <<
<if(rest(model.segments))>
<! requires segmented representation !>
private static final int _serializedATNSegments = <length(model.segments)>;
<model.segments:{segment|private static final String _serializedATNSegment<i0> =
"<segment; wrap={"+<\n><\t>"}>";}; separator="\n">
public static final String _serializedATN = Utils.join(
new String[] {
<model.segments:{segment | _serializedATNSegment<i0>}; separator=",\n">
},
""
);
<else>
<! only one segment, can be inlined !>
public static final String _serializedATN =
"<model.serialized; wrap={"+<\n><\t>"}>";
<endif>
public static final ATN _ATN =
new ATNDeserializer().deserialize(_serializedATN.toCharArray());
static {
_decisionToDFA = new DFA[_ATN.getNumberOfDecisions()];
for (int i = 0; i < _ATN.getNumberOfDecisions(); i++) {
_decisionToDFA[i] = new DFA(_ATN.getDecisionState(i), i);
}
<! org.antlr.v4.tool.DOTGenerator dot = new org.antlr.v4.tool.DOTGenerator(null);!>
<! System.out.println(dot.getDOT(_ATN.decisionToState.get(0), ruleNames, false));!>
<! System.out.println(dot.getDOT(_ATN.ruleToStartState[2], ruleNames, false));!>
}
>>

/** Using a type to init value map, try to init a type; if not in table
* must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<javaTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".java"

```

Found in path(s):

* /opt/cola/permits/1294101088_1649016484.9/0/antlr4-4-7-2-jar/org/antlr/v4/tool/templates/codegen/Java/Java.stg

1.33 junit-5-bill-of-materials 5.7.1

1.33.1 Available under license :

```
import java.io.File
import java.net.URI
```

```
data class License(val name: String, val url: URI, val headerFile: File)
```

```
Apache License
```

```
=====
```

```
_Version 2.0, January 2004_
```

```
_&lt;<https://www.apache.org/licenses/>&gt;_
```

```
### Terms and Conditions for use, reproduction, and distribution
```

```
#### 1. Definitions
```

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and

reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and

****ij**** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a**** distributing or ****b**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2)**** When the Program is Distributed as Source Code:

****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3)**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such

provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here }.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for

the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices

that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate

and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets `[]` replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same printed page as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Open Source Licenses

=====

This product may include a number of subcomponents with separate

copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

[[contributors]]

== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.34 apache-xml-commons 1.4.01

1.34.1 Available under license :

xml-commons/java/external/LICENSE.dom-software.txt \$Id: LICENSE.dom-software.txt 734314 2009-01-14 03:33:27Z mrglavas \$

This license came from: <http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.zip> (COPYRIGHT.html)

W3C SOFTWARE NOTICE AND LICENSE

Copyright 2004 World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved.

The DOM bindings are published under the W3C Software Copyright Notice and License. The software license requires "Notice of any changes or modifications to the W3C files, including the date changes were made." Consequently, modified versions of the DOM bindings must document that they do not conform to the W3C standard; in the case of the IDL definitions, the pragma prefix can no longer be 'w3c.org'; in the case of the Java language binding, the package names can no longer be in the 'org.w3c' package.

Note: The original version of the W3C Software Copyright Notice and License could be found at <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
=====  
== NOTICE file corresponding to section 4(d) of the Apache License, ==  
== Version 2.0, in this case for the Apache xml-commons xml-apis ==  
== distribution. ==  
=====
```

Apache XML Commons XML APIs
Copyright 1999-2009 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- software copyright (c) 2000 World Wide Web Consortium, <http://www.w3.org>

This license came from: <http://www.w3.org/Consortium/Legal/copyright-documents-20021231>

W3C DOCUMENT LICENSE

<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>

Public documents on the W3C site are provided by the copyright holders under the following license. By using and/or copying this document, or the W3C document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [date-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle \$Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) \$
xml-commons/java/external/LICENSE.sax.txt \$Id: LICENSE.sax.txt 225954 2002-01-31 23:26:48Z curcuru \$

This license came from: <http://www.megginson.com/SAX/copying.html>
However please note future versions of SAX may be covered
under <http://saxproject.org/?selected=pd>

This page is now out of date -- see the new SAX site at
<http://www.saxproject.org/> for more up-to-date
releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com

1.35 opentracing-api 0.33.0

1.35.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2016-2019 The OpenTracing Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/tag/StringTag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/Scope.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/tag/IntOrStringTag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/References.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/BinaryInject.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/Format.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/BinaryAdapters.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/TextMapInject.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/Span.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/tag/BooleanTag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/TextMap.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-
jar/io/opentracing/propagation/Binary.java
```


* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/propagation/TextMapInjectAdapter.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/propagation/TextMapAdapter.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/Tracer.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/log/Fields.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/tag/IntTag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/tag/Tag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/tag/Tags.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/propagation/BinaryExtract.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/ScopeManager.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/tag/AbstractTag.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/propagation/TextMapExtract.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/SpanContext.java
* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/io/opentracing/propagation/TextMapExtractAdapter.java

No license file was found, but licenses were detected in source scan.

<!--

Copyright 2016-2019 The OpenTracing Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

or implied. See the License for the specific language governing permissions and limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188637_1654650533.6628933/0/opentracing-api-0-33-0-sources-3-jar/META-INF/maven/io.opentracing/opentracing-api/pom.xml

1.36 bean-validation-api 2.0.1

1.36.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2013 Cognifide Limited

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.37 apache-commons-text 1.9

1.37.1 Available under license :

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Apache Commons Text

Copyright 2014-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

1.38 nimbus-jose-jwt 9.15.2

1.38.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * nimbus-jose-jwt
 *
 * Copyright 2012-2021, Connect2id Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/bc/BouncyCastleFIPSProviderSingleton.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/bc/package-info.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * nimbus-jose-jwt
 *
 * Copyright 2012-2016, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */
```

*/

Found in path(s):

- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/OctetKeyPair.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/gen/JWKGenerator.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/AlgorithmSupportMessage.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/opts/package-info.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/CipherHelper.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/ECDSA.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/produce/JWSSignerFactory.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/gen/ECKeyGenerator.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/JWSSignerOption.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/BaseJWSProvider.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/PEMEncodedKeyParser.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/AESKW.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/AlgorithmParametersHelper.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/RSAKeyUtils.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/ConcatKDF.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/produce/package-info.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/PasswordLookup.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/CompositeKey.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/DeflateHelper.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/opts/UserAuthenticationRequired.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/proc/JWKSecurityContext.java
- * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/util/IntegerOverflowException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSA_OAEP_SHA2.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/LegacyAESGCM.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/LegacyConcatKDF.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWKException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/KeyUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/PBKDF2.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/gen/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/Container.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AESGCMKW.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/CompletableJWSObjectSigning.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/BaseJWEProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/DefaultJOSEObjectTypeVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/KeyUseAndOpsConsistency.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/JSONStringUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AESCBC.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/MACProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/JWTClaimsSetAwareJWSKeySelector.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/ConstantTimeUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/JWKSetWithTimestamp.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/BoundedInputStream.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/ContentCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/gen/OctetSequenceKeyGenerator.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/ActionRequiredForJWSCompletionException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/crypto/impl/ECDSAProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/opts/OptionUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AuthenticatedCipherText.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/PasswordBasedCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/gen/RSAPublicKeyGenerator.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/CurveBasedJWK.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSASSA.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/PRFParams.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/Curve.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AESGCM.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/DirectCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/ECDH.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/CriticalHeaderParamsDeferral.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/AESEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AAD.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/EdDSAProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/opts/AllowWeakRSAKey.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSA_OAEP.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/IOUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/HMAC.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/DefaultJWKSetCache.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSA1_5.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/ECChecks.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/jwk/source/JWKSetCache.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSASSAProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/AESCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/impl/RSACryptoProvider.java
No license file was found, but licenses were detected in source scan.

```
/*  
* nimbus-jose-jwt  
*  
* Copyright 2012-2016, Connect2id Ltd.  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use  
* this file except in compliance with the License. You may obtain a copy of the  
* License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed  
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR  
* CONDITIONS OF ANY KIND, either express or implied. See the License for the  
* specific language governing permissions and limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/RemoteJWKSet.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWEAlgorithm.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSAlgorithm.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/JWKSource.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/SecretJWK.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWEEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/Header.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/Payload.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/util/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/JWTClaimsSetTransformer.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/PlainObject.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/BadJWSException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWKSelector.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/Resource.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/Base64.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/JWTClaimsSetVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/JSONObjectUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/Algorithm.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/KeySourceException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/KeyOperation.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSHeader.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/ECKey.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/SecurityContext.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSObject.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/ConfigurableJWTProcessor.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/JWT.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/PlainHeader.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/ECPParameterTable.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWEDecrypterFactory.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/ByteUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/KeyException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/DateUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/PasswordBasedEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWEKeySelector.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/RSAEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/SignedJWT.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/JWTProcessorConfiguration.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/ECDHDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/ImmutableJWKSet.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/DefaultResourceRetriever.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSSigner.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWKMetadata.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWSKeySelector.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/ECDSASigner.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/factories/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JOSEProcessor.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/ThumbprintUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/BadJOSEException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWK.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/SimpleSecurityContext.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/EncryptedJWT.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JOSEException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/KeyTypeException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/X509CertUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JOSEMatcher.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/source/ImmutableSecret.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/JWTClaimsSet.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/PayloadTransformer.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/KeyConverter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/JWTParser.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/factories/DefaultJWSVerifierFactory.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/OctetSequenceKey.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/EncryptionMethod.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jca/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/RSAPublicKey.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/BadJWEException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/CompressionAlgorithm.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/DefaultJWTClaimsVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/factories/DefaultJWEDecrypterFactory.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/AbstractRestrictedResourceRetriever.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jca/JCAContext.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/DeflateUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/AlgorithmFamily.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/RestrictedResourceRetriever.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/ECDSAVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/X509CertChainUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/jwk/KeyType.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/DirectEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWEObject.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/CommonSEHeader.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/ConfigurableJOSEProcessor.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/RSASSASigner.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/BigIntegerUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/StandardCharset.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JOSEObjectType.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/JSONArrayUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/MACVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/ResourceRetriever.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/AsymmetricJWK.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jca/JCAAware.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/BadJWTException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWEDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/Requirement.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/Base64URL.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/IntegerUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/util/DateUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/MACSigner.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JOSEObject.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/crypto/DirectDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/AESDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWSVerifierFactory.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/ArrayUtils.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JOSEProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWECryptoParts.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/RSADecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/AbstractJWKSelectorWithSource.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/bc/BouncyCastleProviderSingleton.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jca/JCASupport.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWSProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/package-info.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/RemoteKeySourceException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/CriticalHeaderParamsAware.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/KeyLengthException.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jca/JWEJCAContext.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/ClockSkewAware.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/RSASSAVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/KeyUse.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/JWTProcessor.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/PlainJWT.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/X25519Encrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/PasswordBasedDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/JWEProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/JWEHeader.java

No license file was found, but licenses were detected in source scan.

```
/*
 * nimbus-jose-jwt
 *
 * Copyright 2012-2019, Connect2id Ltd.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed
 * under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
 * CONDITIONS OF ANY KIND, either express or implied. See the License for the
 * specific language governing permissions and limitations under the License.
 */
```

Found in path(s):

```
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWSVerificationKeySelector.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWEDecryptionKeySelector.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWKMatcher.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/DefaultJOSEProcessor.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JOSEProcessorConfiguration.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/proc/JWSAlgorithmFamilyJWSKeySelector.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/ECDHEncrypter.java
 * /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jwt/proc/DefaultJWTProcessor.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * nimbus-jose-jwt
 *
 * Copyright 2012-2019, Connect2id Ltd and contributors.
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use
 * this file except in compliance with the License. You may obtain a copy of the
 * License at
 *
 *
```

* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/ECDHCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/proc/JOSEObjectTypeVerifier.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/jwk/source/JWKSecurityContextJWKSet.java

No license file was found, but licenses were detected in source scan.

/*

* nimbus-jose-jwt
*
* Copyright 2012-2021, Connect2id Ltd and contributors.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use
* this file except in compliance with the License. You may obtain a copy of the
* License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed
* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/ECDH1PUX25519Decrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/ECDH1PUX25519Encrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/ECDH1PUDecrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/ECDH1PUCryptoProvider.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/ECDH1PUEncrypter.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-jar/com/nimbusds/jose/crypto/impl/ECDH1PU.java
* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/crypto/impl/XC20P.java

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-

jar/com/nimbusds/jose/mint/package-info.java

No license file was found, but licenses were detected in source scan.

/*

* nimbus-jose-jwt

*

* Copyright 2012-2018, Connect2id Ltd.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use

* this file except in compliance with the License. You may obtain a copy of the

* License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed

* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR

* CONDITIONS OF ANY KIND, either express or implied. See the License for the

* specific language governing permissions and limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/X25519Decrypter.java

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/Ed25519Signer.java

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/JWKSet.java

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/Ed25519Verifier.java

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/util/Base64Codec.java

No license file was found, but licenses were detected in source scan.

/*

* nimbus-jose-jwt

*

* Copyright 2012-2018, Connect2id Ltd and contributors.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use

* this file except in compliance with the License. You may obtain a copy of the

* License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed

* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR

* CONDITIONS OF ANY KIND, either express or implied. See the License for the
* specific language governing permissions and limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/jwk/gen/OctetKeyPairGenerator.java

No license file was found, but licenses were detected in source scan.

/*

* nimbus-jose-jwt

*

* Copyright 2012-2020, Connect2id Ltd.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use

* this file except in compliance with the License. You may obtain a copy of the

* License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed

* under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR

* CONDITIONS OF ANY KIND, either express or implied. See the License for the

* specific language governing permissions and limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188529_1654650478.0218625/0/nimbus-jose-jwt-9-15-2-sources-
jar/com/nimbusds/jose/crypto/factories/DefaultJWSSignerFactory.java

1.39 azure-java-sdk-bom-(bill-of-materials)

1.1.1

1.39.1 Available under license :

// Copyright (c) Microsoft Corporation. All rights reserved.

// Licensed under the MIT License.

// Code generated by Microsoft (R) AutoRest Code Generator.

The MIT License (MIT)

Copyright (c) 2015 Microsoft

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is

furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.40 fabric8-::-kubernetes-model-::-common

4.10.3

1.40.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.41 jakarta-ws-rs-api 2.1.6

1.41.1 Available under license :

Notices for the Jakarta RESTful Web Services Project

This content is produced and maintained by the **Jakarta RESTful Web Services** project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jaxrs>

Trademarks

Jakarta RESTful Web Services is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed

source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jaxrs-api>

Third-party Content

This project leverages the following third party content.

javaee-api (7.0)

* License: Apache-2.0 AND W3C

JUnit (4.11)

* License: Common Public License 1.0

Mockito (2.16.0)

* Project: <http://site.mockito.org>

* Source: <https://github.com/mockito/mockito/releases/tag/v2.16.0>

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE

PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR

PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement

Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License.

(Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any

associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you

could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free

software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU

General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands ``show w'` and ``show c'` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ``show w'` and ``show c'`; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

```
## CLASSPATH EXCEPTION
```

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you

permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.42 okio 2.5.0

1.42.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
*/opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/ByteString.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
```

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-

jar/commonMain/okio/SegmentedByteString.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/ForwardingTimeout.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/SegmentedByteString.kt
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2016 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Pipe.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Options.kt
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2019 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-

jar/commonMain/okio/BufferedSource.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Buffer.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Source.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Timeout.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Sink.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/BufferedSink.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Okio.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/Buffer.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/-
DeprecatedUtf8.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/-
DeprecatedUpgrade.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Throttler.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/-
DeprecatedOkio.kt
* /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/PeekSource.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");

- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/Segment.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/SegmentPool.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/ForwardingSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/BufferedSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/AsyncTimeout.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/GzipSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/JvmOkio.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/RealBufferedSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/ForwardingSink.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Buffer.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/InflaterSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/BufferedSink.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Sink.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/DeflaterSink.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/GzipSink.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Timeout.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/Source.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/RealBufferedSink.kt

No license file was found, but licenses were detected in source scan.

- /*
- * Copyright (C) 2019 Square, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/RealBufferedSource.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/SegmentedByteString.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/RealBufferedSink.kt

No license file was found, but licenses were detected in source scan.

/*

- * Copyright (C) 2018 Square, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/-Util.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/ByteString.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/-Platform.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/ByteString.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/jvmMain/okio/-Platform.kt
- * /opt/cola/permits/1337069344_1654119302.7043238/0/okio-2-5-0-sources-4-jar/commonMain/okio/internal/-Utf8.kt

1.43 commons-logging 1.1.1

1.43.1 Available under license :

/*

- * Licensed to the Apache Software Foundation (ASF) under one
- * or more contributor license agreements. See the NOTICE file
- * distributed with this work for additional information

* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Apache Commons Logging

Copyright 2003-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.44 jackson-core 2.13.2

1.44.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0 To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.45 jackson-xc 2.13.2

1.45.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.

It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

This copy of Jackson JSON processor `jackson-module-jaxb-annotations` module is licensed under the
Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.46 fabric8-::-openshift-model 4.10.3

1.46.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.47 jackson-module-parameter-names 2.13.2

1.47.1 Available under license :

Apache-2.0

1.48 metrics-health-checks 4.1.17

1.48.1 Available under license :

Apache-2.0

1.49 junit-platform-junit-platform-engine 1.7.2

1.49.1 Available under license :

```
import java.io.File  
import java.net.URI
```

```
data class License(val name: String, val url: URI, val headerFile: File)
```

```
Apache License
```

```
=====
```

```
_Version 2.0, January 2004_
```

```
_&lt;<https://www.apache.org/licenses/>&gt;_
```

```
### Terms and Conditions for use, reproduction, and distribution
```

```
#### 1. Definitions
```

License shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

Licensors shall mean the copyright owner or entity authorized by the copyright

owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work,

provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **j)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* ****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that

such license:

* **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)*** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY

APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All

rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or

translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was

submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets `[]` replaced with your own

identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same printed page as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

[[contributors]]

== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.50 protocol-buffer-java-api 3.19.4

1.50.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Copyright 2008 Google Inc. All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// * Neither the name of Google Inc. nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/compiler/plugin.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/wrappers.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/any.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/type.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/timestamp.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/empty.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/source_context.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/duration.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/descriptor.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/struct.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/field_mask.proto
- * /opt/cola/permits/1319213656_1651767399.38/0/protobuf-java-shaded-3-19-4-1-jar/google/protobuf/api.proto

1.51 pact-jvm-core-matchers 4.0.10

1.51.1 Available under license :

Apache-2.0

1.52 google-guice 4.0

1.52.1 Available under license :

Struts 2 Core

Copyright 2000-2010 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).
Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright statements and notices. Redistributions must also contain a copy of this document.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com.
4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
5. Due credit should be given to the DOM4J Project - <http://www.dom4j.org>

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

ASM: a very small and fast Java bytecode manipulation framework

Copyright (c) 2000,2002,2003 INRIA, France Telecom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Jakarta Commons IO

Copyright 2001-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Specification: JSR-000220 Enterprise JavaBeans v.3.0 ("Specification")

Version: 3.0

Status: Final Release

Release: 8 May 2006

Copyright 2006 SUN MICROSYSTEMS, INC.

4150 Network Circle, Santa Clara, California 95054, U.S.A

All rights reserved.

LIMITED LICENSE GRANTS

1. License for Evaluation Purposes. Sun hereby grants you a fully-paid, non-exclusive, non-transferable, worldwide, limited license (without the right to sublicense), under Sun's applicable intellectual property rights to view, download, use and reproduce the Specification only for the purpose of internal evaluation. This includes (i) developing applications intended to run on an implementation of the Specification, provided that such applications do not themselves implement any portion(s) of the Specification, and (ii) discussing the Specification with any third party; and (iii) excerpting brief portions of the Specification in oral or written communications which discuss the Specification provided that such excerpts do not in the aggregate constitute a significant portion of the Specification.

2. License for the Distribution of Compliant Implementations. Sun also grants you a perpetual, nonexclusive, non-transferable, worldwide, fully paid-up, royalty free, limited license (without the right to sublicense) under any applicable copyrights or, subject to the provisions of subsection 4 below, patent rights it may have covering the Specification to create and/or distribute an Independent Implementation of the Specification that: (a) fully implements the Specification including all its required interfaces and functionality; (b) does not modify, subset, superset or otherwise extend the Licensor Name Space, or include

any public or protected packages, classes, Java interfaces, fields or methods within the Licensor Name Space other than those required/authorized by the Specification or Specifications being implemented; and (c) passes the Technology Compatibility Kit (including satisfying the requirements of the applicable TCK Users Guide) for such Specification ("Compliant Implementation"). In addition, the foregoing license is expressly conditioned on your not acting outside its scope. No license is granted hereunder for any other purpose (including, for example, modifying the Specification, other than to the extent of your fair use rights, or distributing the Specification to third parties). Also, no right, title, or interest in or to any trademarks, service marks, or trade names of Sun or Sun's licensors is granted hereunder. Java, and Java-related logos, marks and names are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

3. Pass-through Conditions. You need not include limitations (a)-(c) from the previous paragraph or any other particular "pass through" requirements in any license You grant concerning the use of your Independent Implementation or products derived from it. However, except with respect to Independent Implementations (and products derived from them) that satisfy limitations (a)-(c) from the previous paragraph, You may neither: (a) grant or otherwise pass through to your licensees any licenses under Sun's applicable intellectual property rights; nor (b) authorize your licensees to make any claims concerning their implementations compliance with the Specification in question.

4. Reciprocity Concerning Patent Licenses.

a. With respect to any patent claims covered by the license granted under subparagraph 2 above that would be infringed by all technically feasible implementations of the Specification, such license is conditioned upon your offering on fair, reasonable and non-discriminatory terms, to any party seeking it from You, a perpetual, non-exclusive, non-transferable, worldwide license under Your patent rights which are or would be infringed by all technically feasible implementations of the Specification to develop, distribute and use a Compliant Implementation.

b With respect to any patent claims owned by Sun and covered by the license granted under subparagraph 2, whether or not their infringement can be avoided in a technically feasible manner when implementing the Specification, such license shall terminate with respect to such claims if You initiate a claim against Sun that it has, in the course of performing its responsibilities as the Specification Lead, induced any other entity to infringe Your patent rights.

c Also with respect to any patent claims owned by Sun and covered by the license granted under subparagraph 2 above, where the infringement of such claims can be avoided in a technically feasible manner when implementing the Specification such license, with respect to such claims, shall terminate if You initiate a claim against Sun that its making, having made, using, offering to sell, selling or importing a Compliant Implementation infringes Your patent rights.

5. Definitions. For the purposes of this Agreement: "Independent Implementation" shall mean an implementation of the Specification that neither derives from any of Sun's source code or binary code materials nor, except with an appropriate and separate license from Sun, includes any of Sun's source code or binary code materials; "Licensor Name Space" shall mean the public class or interface declarations whose names begin with "java", "javax", "com.sun" or their equivalents in any subsequent naming convention adopted by Sun through the Java Community Process, or any recognized successors or replacements thereof; and "Technology Compatibility Kit" or "TCK" shall mean the test suite and accompanying TCK Users Guide provided by Sun which corresponds to the Specification and that was available either (i) from Sun 120 days before the first release of Your Independent Implementation that allows its use for commercial purposes, or (ii) more recently than 120 days from such release but against which You elect to test Your implementation of the Specification.

This Agreement will terminate immediately without notice from Sun if you breach the Agreement or act outside the scope of the licenses granted above.

DISCLAIMER OF WARRANTIES

THE SPECIFICATION IS PROVIDED "AS IS". SUN MAKES NO REPRESENTATIONS OR WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT (INCLUDING AS A CONSEQUENCE OF ANY PRACTICE OR IMPLEMENTATION OF THE SPECIFICATION), OR THAT THE CONTENTS OF THE SPECIFICATION ARE SUITABLE FOR ANY PURPOSE. This document does not represent any commitment to release or implement any portion of the Specification in any product. In addition, the Specification could include technical inaccuracies or typographical errors.

LIMITATION OF LIABILITY

TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION, LOST REVENUE, PROFITS OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED IN ANY WAY TO YOUR HAVING, IMPLEMENTING OR OTHERWISE USING THE SPECIFICATION, EVEN IF SUN AND/OR ITS LICENSORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You will indemnify, hold harmless, and defend Sun and its licensors from any claims arising or resulting from: (i) your use of the Specification; (ii) the use or distribution of your Java application, applet and/or implementation; and/or (iii) any claims that later versions or releases of any Specification furnished to you are incompatible with the Specification provided to you under this license.

RESTRICTED RIGHTS LEGEND

U.S. Government: If this Specification is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in the Software and accompanying documentation shall be only as set forth in this license; this is in accordance with 48 C.F.R. 227.7201 through 227.7202-4 (for Department of Defense (DoD) acquisitions) and with 48 C.F.R. 2.101 and 12.212 (for non-DoD acquisitions).

REPORT

If you provide Sun with any comments or suggestions concerning the Specification ("Feedback"), you hereby: (i) agree that such Feedback is provided on a non-proprietary and non-confidential basis, and (ii) grant Sun a perpetual, non-exclusive, worldwide, fully paid-up, irrevocable license, with the right to sublicense through multiple levels of sublicensees, to incorporate, disclose, and use without limitation the Feedback for any purpose.

GENERAL TERMS

Any action related to this Agreement will be governed by California law and controlling U.S. federal law. The U.N. Convention for the International Sale of Goods and the choice of law rules of any jurisdiction will not apply.

The Specification is subject to U.S. export control laws and may be subject to export or import regulations in other countries. Licensee agrees to comply strictly with all such laws and regulations and acknowledges that it has the responsibility to obtain such licenses to export, re-export or import as may be required after delivery to Licensee.

This Agreement is the parties' entire agreement relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, conditions, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification to this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

Rev. April, 2006

Sun/Final/Full

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).
Apache Struts
Copyright 2000-2007 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).
domTT (<http://www.mojavelinux.com/projects/domtooltip/>).

The binary distributions includes the following third party software:
FreeMarker (<http://freemarker.org/>).
OGNL (<http://www.opensymphony.com/ognl/>).
XWork (<http://www.opensymphony.com/xwork/>).

```
=====
== NOTICE file corresponding to the section 4 d of           ==
== the Apache License, Version 2.0,                          ==
== in this case for the Apache Ant distribution.              ==
=====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

This product includes also software developed by :
- the W3C consortium (<http://www.w3c.org/>) ,
- the SAX project (<http://www.saxproject.org>)

Please read the different LICENSE files present in the root directory of
this distribution.

```
/* =====
* The OpenSymphony Software License, Version 1.1
*
* (this license is derived and fully compatible with the Apache Software
* License - see http://www.apache.org/LICENSE.txt)
*
* Copyright (c) 2001-2004 The OpenSymphony Group. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
```

*
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * OpenSymphony Group (<http://www.opensymphony.com/>)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "OpenSymphony" and "The OpenSymphony Group"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact license@opensymphony.com .
 *
 * 5. Products derived from this software may not be called "OpenSymphony"
 * or "XWork", nor may "OpenSymphony" or "XWork" appear in their
 * name, without prior written permission of the OpenSymphony Group.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 */

Apache Felix Main
 Copyright 2010 The Apache Software Foundation

I. Included Software

This product includes software developed at
 The Apache Software Foundation (<http://www.apache.org/>).
 Licensed under the Apache License 2.0.

This product includes software developed at
 The OSGi Alliance (<http://www.osgi.org/>).
 Copyright (c) OSGi Alliance (2000, 2009).
 Licensed under the Apache License 2.0.

II. Used Software

This product uses software developed at
The OSGi Alliance (<http://www.osgi.org/>).
Copyright (c) OSGi Alliance (2000, 2009).
Licensed under the Apache License 2.0.

This product uses software developed at
The Codehaus (<http://www.codehaus.org>)
Licensed under the Apache License 2.0.

III. License Summary

- Apache License 2.0

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Lang

Copyright 2001-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).
Apache Commons FileUpload
Copyright 2002-2008 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

/*

* Apache License
* Version 2.0, January 2004
* <http://www.apache.org/licenses/>

* TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

*

* 1. Definitions.

*

* "License" shall mean the terms and conditions for use, reproduction,
* and distribution as defined by Sections 1 through 9 of this document.

*

* "Licensor" shall mean the copyright owner or entity authorized by
* the copyright owner that is granting the License.

*

* "Legal Entity" shall mean the union of the acting entity and all
* other entities that control, are controlled by, or are under common
* control with that entity. For the purposes of this definition,
* "control" means (i) the power, direct or indirect, to cause the
* direction or management of such entity, whether by contract or
* otherwise, or (ii) ownership of fifty percent (50%) or more of the
* outstanding shares, or (iii) beneficial ownership of such entity.
*

* "You" (or "Your") shall mean an individual or Legal Entity
* exercising permissions granted by this License.
*

* "Source" form shall mean the preferred form for making modifications,
* including but not limited to software source code, documentation
* source, and configuration files.
*

* "Object" form shall mean any form resulting from mechanical
* transformation or translation of a Source form, including but
* not limited to compiled object code, generated documentation,
* and conversions to other media types.
*

* "Work" shall mean the work of authorship, whether in Source or
* Object form, made available under the License, as indicated by a
* copyright notice that is included in or attached to the work
* (an example is provided in the Appendix below).
*

* "Derivative Works" shall mean any work, whether in Source or Object
* form, that is based on (or derived from) the Work and for which the
* editorial revisions, annotations, elaborations, or other modifications
* represent, as a whole, an original work of authorship. For the purposes
* of this License, Derivative Works shall not include works that remain
* separable from, or merely link (or bind by name) to the interfaces of,
* the Work and Derivative Works thereof.
*

* "Contribution" shall mean any work of authorship, including
* the original version of the Work and any modifications or additions
* to that Work or Derivative Works thereof, that is intentionally
* submitted to Licensor for inclusion in the Work by the copyright owner
* or by an individual or Legal Entity authorized to submit on behalf of
* the copyright owner. For the purposes of this definition, "submitted"
* means any form of electronic, verbal, or written communication sent
* to the Licensor or its representatives, including but not limited to
* communication on electronic mailing lists, source code control systems,
* and issue tracking systems that are managed by, or on behalf of, the
* Licensor for the purpose of discussing and improving the Work, but
* excluding communication that is conspicuously marked or otherwise
* designated in writing by the copyright owner as "Not a Contribution."
*

* "Contributor" shall mean Licensor and any individual or Legal Entity

* on behalf of whom a Contribution has been received by Licensor and
* subsequently incorporated within the Work.

*

* 2. Grant of Copyright License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* copyright license to reproduce, prepare Derivative Works of,
* publicly display, publicly perform, sublicense, and distribute the
* Work and such Derivative Works in Source or Object form.

*

* 3. Grant of Patent License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* (except as stated in this section) patent license to make, have made,
* use, offer to sell, sell, import, and otherwise transfer the Work,
* where such license applies only to those patent claims licensable
* by such Contributor that are necessarily infringed by their
* Contribution(s) alone or by combination of their Contribution(s)
* with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.

*

* 4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:

*

* (a) You must give any other recipients of the Work or
* Derivative Works a copy of this License; and

*

* (b) You must cause any modified files to carry prominent notices
* stating that You changed the files; and

*

* (c) You must retain, in the Source form of any Derivative Works
* that You distribute, all copyright, patent, trademark, and
* attribution notices from the Source form of the Work,
* excluding those notices that do not pertain to any part of
* the Derivative Works; and

*

* (d) If the Work includes a "NOTICE" text file as part of its
* distribution, then any Derivative Works that You distribute must
* include a readable copy of the attribution notices contained
* within such NOTICE file, excluding those notices that do not
* pertain to any part of the Derivative Works, in at least one

* of the following places: within a NOTICE text file distributed
* as part of the Derivative Works; within the Source form or
* documentation, if provided along with the Derivative Works; or,
* within a display generated by the Derivative Works, if and
* wherever such third-party notices normally appear. The contents
* of the NOTICE file are for informational purposes only and
* do not modify the License. You may add Your own attribution
* notices within Derivative Works that You distribute, alongside
* or as an addendum to the NOTICE text from the Work, provided
* that such additional attribution notices cannot be construed
* as modifying the License.

*
* You may add Your own copyright statement to Your modifications and
* may provide additional or different license terms and conditions
* for use, reproduction, or distribution of Your modifications, or
* for any such Derivative Works as a whole, provided Your use,
* reproduction, and distribution of the Work otherwise complies with
* the conditions stated in this License.

*
* 5. Submission of Contributions. Unless You explicitly state otherwise,
* any Contribution intentionally submitted for inclusion in the Work
* by You to the Licensor shall be under the terms and conditions of
* this License, without any additional terms or conditions.
* Notwithstanding the above, nothing herein shall supersede or modify
* the terms of any separate license agreement you may have executed
* with Licensor regarding such Contributions.

*
* 6. Trademarks. This License does not grant permission to use the trade
* names, trademarks, service marks, or product names of the Licensor,
* except as required for reasonable and customary use in describing the
* origin of the Work and reproducing the content of the NOTICE file.

*
* 7. Disclaimer of Warranty. Unless required by applicable law or
* agreed to in writing, Licensor provides the Work (and each
* Contributor provides its Contributions) on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
* implied, including, without limitation, any warranties or conditions
* of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
* PARTICULAR PURPOSE. You are solely responsible for determining the
* appropriateness of using or redistributing the Work and assume any
* risks associated with Your exercise of permissions under this License.

*
* 8. Limitation of Liability. In no event and under no legal theory,
* whether in tort (including negligence), contract, or otherwise,
* unless required by applicable law (such as deliberate and grossly
* negligent acts) or agreed to in writing, shall any Contributor be
* liable to You for damages, including any direct, indirect, special,
* incidental, or consequential damages of any character arising as a

* result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill,
* work stoppage, computer failure or malfunction, or any and all
* other commercial damages or losses), even if such Contributor
* has been advised of the possibility of such damages.

*
* 9. Accepting Warranty or Additional Liability. While redistributing
* the Work or Derivative Works thereof, You may choose to offer,
* and charge a fee for, acceptance of support, warranty, indemnity,
* or other liability obligations and/or rights consistent with this
* License. However, in accepting such obligations, You may act only
* on Your own behalf and on Your sole responsibility, not on behalf
* of any other Contributor, and only if You agree to indemnify,
* defend, and hold each Contributor harmless for any liability
* incurred by, or claims asserted against, such Contributor by reason
* of your accepting any such warranty or additional liability.

* END OF TERMS AND CONDITIONS

* APPENDIX: How to apply the Apache License to your work.

* To apply the Apache License to your work, attach the following
* boilerplate notice, with the fields enclosed by brackets "[]"
* replaced with your own identifying information. (Don't include
* the brackets!) The text should be enclosed in the appropriate
* comment syntax for the file format. We also recommend that a
* file or class name and description of purpose be included on the
* same "printed page" as the copyright notice for easier
* identification within third-party archives.

* Copyright [yyyy] [name of copyright owner]

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

* <http://www.apache.org/licenses/LICENSE-2.0>

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and
* limitations under the License.

*/

FreeMarker 1.x was released under the LGPL license. Later, by community
consensus, we have switched over to a BSD-style license. As of FreeMarker
2.2pre1, the original author, Benjamin Geer, has relinquished the copyright in
behalf of Visigoth Software Society. The current copyright holder is the

* Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (<http://www.apache.org/>)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation", "Jakarta-Oro"
 * must not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache"
 * or "Jakarta-Oro", nor may "Apache" or "Jakarta-Oro" appear in their
 * name, without prior written permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * =====
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Apache Software Foundation. For more
 * information on the Apache Software Foundation, please see
 * <http://www.apache.org/>.
 */

XWork: Core
Copyright 2000-2010 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

/* =====

* The OpenSymphony Software License, Version 1.1
*
* (this license is derived and fully compatible with the Apache Software
* License - see <http://www.apache.org/LICENSE.txt>)
*
* Copyright (c) 2001-2004 The OpenSymphony Group. All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* OpenSymphony Group (<http://www.opensymphony.com/>)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "OpenSymphony" and "The OpenSymphony Group"
* must not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact license@opensymphony.com .
*
* 5. Products derived from this software may not be called "OpenSymphony"
* or "OGNL", nor may "OpenSymphony" or "OGNL" appear in their
* name, without prior written permission of the OpenSymphony Group.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT

* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* =====
*/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.53 google-android-annotations-library

4.1.1.4

1.53.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2012 The Android Open Source Project
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1014842714_1591897072.99/0/annotations-4-1-1-4-sources-
jar/android/annotation/SuppressLint.java
* /opt/ws_local/PERMITS_SQL/1014842714_1591897072.99/0/annotations-4-1-1-4-sources-
jar/android/annotation/TargetApi.java
```

1.54 java-classmate classmate-1.5.1

1.54.1 Available under license :

Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Java ClassMate library is licensed under Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.55 fabric8-::-kubernetes-model-::-metrics

4.10.3

1.55.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.56 config 5.5.1

1.56.1 Available under license :

MIT License

Copyright (c) 2020 Netlify <team@netlify.com>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.57 apache-commons-lang 3.4

1.57.1 Available under license :

Apache Commons Lang

Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

This product includes software from the Spring Framework,
under the Apache License 2.0 (see: `StringUtils.containsWhitespace()`)

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.58 aop-alliance 1.0

1.58.1 Available under license :

all the source code provided by AOP Alliance is Public Domain.

1.59 junit-platform-junit-platform-engine 1.7.1

1.59.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **j)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* ****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that

such license:

* **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)*** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY

APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All

rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.60 jackson-datatype-jdk8 2.13.2

1.60.1 Available under license :

Apache-2.0

1.61 commons-codec 1.13

1.61.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Codec

Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from <http://aspell.net/test/orig/batch0.tab>.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at <http://stevemorse.org/phoneticinfo.htm>
with permission from the original authors.

Original source copyright:

Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

1.62 google-api-grpc-proto-google-common-protos 2.0.1

1.62.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");  
// you may not use this file except in compliance with the License.  
// You may obtain a copy of the License at  
// http://www.apache.org/licenses/LICENSE-2.0  
// distributed under the License is distributed on an "AS IS" BASIS,  
// agreement (which includes confidentiality provisions). These features may
```

Found in path(s):

```
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/launch_stage.proto
```

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");  
// you may not use this file except in compliance with the License.  
// You may obtain a copy of the License at  
// http://www.apache.org/licenses/LICENSE-2.0  
// distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/api/label.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-  
jar/google/type/postal_address.proto  
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
```

jar/google/type/datetime.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/source_info.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/quaternion.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/field_behavior.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/latlng.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/httpbody.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/money.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/logging/type/http_request.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/calendar_period.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/rpc/code.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/geo/type/viewport.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/client.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/longrunning/operations.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/billing.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/backend.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/consumer.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/fraction.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/system_parameter.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/rpc/status.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/documentation.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/http.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/rpc/context/attribute_context.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/quota.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/date.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-

jar/google/api/logging.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/cloud/audit/audit_log.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/distribution.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/config_change.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/metric.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/dayofweek.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/endpoint.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/usage.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/auth.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/expr.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/log.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/annotations.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/service.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/monitoring.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/resource.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/logging/type/log_severity.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/color.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/rpc/error_details.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/context.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/type/timeofday.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/control.proto
* /opt/cola/permits/1337069512_1654218326.7907567/0/proto-google-common-protos-2-0-1-1-
jar/google/api/monitored_resource.proto

1.63 wire-multiplatform-runtime- (experimental) 3.2.2

1.63.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Copyright 2008 Google Inc. All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// * Neither the name of Google Inc. nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1337070384_1654202046.8440835/0/wire-schema-3-2-2-jar/google/protobuf/descriptor.proto  
* /opt/cola/permits/1337070384_1654202046.8440835/0/wire-schema-3-2-2-jar/google/protobuf/any.proto
```

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2019 Square Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1337070384_1654202046.8440835/0/wire-schema-3-2-2-jar/wire/extensions.proto
```

1.64 bean-validation-api 2.0.2

1.64.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent

to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work,

excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any

risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2013 Cognifide Limited

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.65 io-grpc-grpc-protobuf-lite 1.39.0

1.65.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2014 The gRPC Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-  
jar/io/grpc/protobuf-lite/ProtoLiteUtils.java
```

```
* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-  
jar/io/grpc/protobuf-lite/ProtoInputStream.java
```

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017 The gRPC Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1183892729_1655096107.2169387/0/grpc-protobuf-lite-1-39-0-sources-jar/io/grpc/protobuf-lite/package-info.java

1.66 dropwizard-lifecycle-support 2.0.18

1.66.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.67 jul-to-slf4j-bridge 1.7.30

1.67.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2004-2013 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).
Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.68 metrics-integration-with-jmx 4.1.17

1.68.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.codahale.metrics.jmx

Build-Jdk-Spec: 1.8
Bundle-Description: A set of classes which allow you to report metrics via JMX.
Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>
Bundle-ManifestVersion: 2
Bundle-Name: Metrics Integration with JMX
Bundle-SymbolicName: io.dropwizard.metrics.jmx
Bundle-Version: 4.1.17
Created-By: Apache Maven Bundle Plugin
Export-Package: com.codahale.metrics.jmx;uses:="com.codahale.metrics,javax.management";version="4.1.17"
Implementation-Title: Metrics Integration with JMX
Implementation-Version: 4.1.17
Import-Package: com.codahale.metrics;version="[4.1,5)",javax.management,org.slf4j;version="[1.6.0,2.0.0)"
Private-Package: com.codahale.metrics.jmx
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188483_1654734106.0939188/0/metrics-jmx-4-1-17-jar/META-INF/MANIFEST.MF

1.69 jackson-module-afterburner 2.13.2

1.69.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.

It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components (as well their dependencies) may be licensed under different licenses.

To find the details that apply to this artifact see the accompanying LICENSE file.

For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

This copy of Jackson JSON processor `jackson-module-afterburner` module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Additional licensing information exists for following 3rd party library dependencies

ASM

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.70 opentelemetry---

io.opentelemetry:opentelemetry-proto 1.6.0-

alpha

1.70.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");  
// you may not use this file except in compliance with the License.  
// You may obtain a copy of the License at  
// http://www.apache.org/licenses/LICENSE-2.0  
// distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/collector/logs/v1/logs_service.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/collector/metrics/v1/metrics_service.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/collector/trace/v1/trace_service.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/trace/v1/trace_config.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/metrics/experimental/metrics_config_service.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/trace/v1/trace.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/logs/v1/logs.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/metrics/v1/metrics.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/resource/v1/resource.proto  
* /opt/cola/permits/1310721520_1650882367.2/0/opentelemetry-proto-1-6-0-alpha-  
jar/opentelemetry/proto/common/v1/common.proto
```

1.71 wire-schema 3.6.0

1.71.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2014 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0
```

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/FieldElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/OneOfElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/EnumConstantElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/RpcElement.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/OptionReader.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Loader.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/jvmMain/com/squareup/wire/schema/internal/UtilJVM.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/FileLinker.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/jvmMain/com/squareup/wire/schema/CoreLoader.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/SemVer.kt

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/SyntaxReader.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/EnclosingType.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Reserved.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/ReservedElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/internal/parser/GroupElement.kt

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2013 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/EnumElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/ServiceElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/ProtoParser.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/ProtoFileElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/TypeElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/ExtendElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/ExtensionsElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/Util.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/OptionElement.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/parser/MessageElement.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2015 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 *
```


- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/MessageType.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/MarkSet.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/PruningRules.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/ProtoMember.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Rpc.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/OneOf.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Pruner.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Extend.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Options.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/ProtoType.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Field.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/EnumType.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/EnumConstant.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Location.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/ProtoFile.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Extensions.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Schema.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Service.kt
- * /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-jar/commonMain/com/squareup/wire/schema/Linker.kt

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/SchemaException.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/SchemaProtoAdapterFactory.kt
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/Type.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (C) 2020 Square, Inc.
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/jvmMain/com/squareup/wire/schema/EmittingRules.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/jvmMain/com/squareup/wire/schema/ClaimedDefinitions.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/CoreLoader.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/jvmMain/com/squareup/wire/schema/Multimap.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/SyntaxRules.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/ErrorCollector.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/Multimap.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/jvmMain/com/squareup/wire/schema/internal/JvmLanguages.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/DagChecker.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/internal/TypeMover.kt
```

```
* /opt/cola/permits/1340031715_1654618469.2355554/0/wire-schema-3-6-0-sources-1-
jar/commonMain/com/squareup/wire/schema/CycleChecker.kt
```

1.72 micronaut-kafka 4.2.0

1.72.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
# Licensed under the Apache License, Version 2.0 (the "License");  
# you may not use this file except in compliance with the License.  
# You may obtain a copy of the License at  
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1340188572_1654650458.7274003/0/micronaut-kafka-4-2-0-jar/META-INF/native-  
image/io.micronaut.kafka/micronaut-kafka/native-image.properties
```

1.73 fabric8-::-kubernetes-model-::-rbac

4.10.3

1.73.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.74 jacoco v0.8.6

1.74.1 Available under license :

License

=====

Copyright (c) 2009, 2020 Mountainminds GmbH & Co. KG and Contributors

The JaCoCo Java Code Coverage Library and all included documentation is made available by Mountainminds GmbH & Co. KG, Munich. Except indicated below, the Content is provided to you under the terms and conditions of the Eclipse Public License Version 2.0 ("EPL"). A copy of the EPL is available at [\[https://www.eclipse.org/legal/epl-2.0/\]](https://www.eclipse.org/legal/epl-2.0/)(<https://www.eclipse.org/legal/epl-2.0/>).

Please visit

<http://www.jacoco.org/jacoco/trunk/doc/license.html>
for the complete license information including third party licenses and trademarks.

1.75 micronaut-rxjava-3 2.2.1

1.75.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedComponent.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedCompletableObserver.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/converters/RxJava2ToRxJava3ConverterRegistrar.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedParallelFlowable.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedSingleObserver.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedConnectableFlowable.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedObserver.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedObservable.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/converters/RxJava3ConverterRegistrar.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedCompletable.java
* /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-
jar/io/micronaut/rxjava3/instrument/RxInstrumentedMaybeObserver.java
```


- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedConnectableObservable.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxJava3Instrumentation.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedWrappers.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedFlowableSubscriber.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedFlowable.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedSubscriber.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedMaybe.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumentedSingle.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/RxInstrumenterFactory.java
- * /opt/cola/permits/1331474197_1653513200.9120781/0/micronaut-rxjava3-2-2-1-sources-jar/io/micronaut/rxjava3/instrument/package-info.java

1.76 kotlinx-coroutines-io-jvm 0.1.16

1.76.1 Available under license :

No license file was found, but licenses were detected in source scan.

```

/*
 * Copyright 2016-2017 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */

```

Found in path(s):

- * /opt/cola/permits/1162864466_1620924146.81/0/kotlinx-coroutines-io-jvm-0-1-16-sources-1-jar/commonMain/kotlinx/coroutines/io/ByteChannelCtor.kt

1.77 micronaut-coherence 3.0.1

1.77.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
#  
# Copyright 2017-2020 original authors  
#  
# Licensed under the Apache License, Version 2.0 (the "License");  
# you may not use this file except in compliance with the License.  
# You may obtain a copy of the License at  
#  
# https://www.apache.org/licenses/LICENSE-2.0  
#  
# Unless required by applicable law or agreed to in writing, software  
# distributed under the License is distributed on an "AS IS" BASIS,  
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
# See the License for the specific language governing permissions and  
# limitations under the License.  
#
```

io.micronaut.coherence.config.MicronautSystemPropertyResolver

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/META-INF/services/com.tangosol.coherence.config.SystemPropertyResolver  
No license file was found, but licenses were detected in source scan.
```

```
/*  
* Copyright 2017-2020 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/config/CoherenceDefaultProperties.java
```

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/namespace/BeanBuilder.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/package-info.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/CoherenceContext.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/grpc/GrpcServerBuilder.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/SessionType.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/namespace/BeanProcessor.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/SubscriberExecutorServiceConfig.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/config/MicronautSystemPropertyResolver.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/namespace/package-info.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/package-info.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/SessionConfigurationProvider.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/AbstractSessionConfigurationBean.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/namespace/MicronautNamespaceHandler.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/TopicKey.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/JsonSerializerFactory.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/grpc/GrpcServerConfigurer.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/grpc/package-info.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/SerializerFactories.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/SessionConfigurationBean.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/InjectorImpl.java

No license file was found, but licenses were detected in source scan.

```
#  
# Copyright 2017-2020 original authors  
#  
# Licensed under the Apache License, Version 2.0 (the "License");  
# you may not use this file except in compliance with the License.  
# You may obtain a copy of the License at
```

```
#
# https://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.
#
```

io.micronaut.coherence.grpc.GrpcServerBuilder\$GrpcServerBuilderProviderBean

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/META-INF/services/com.oracle.coherence.grpc.proxy.GrpcServerBuilderProvider
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright 2017-2021 original authors
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* https://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/FilterFactories.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/ExtractorFactory.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Updated.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/WhereFilter.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/EventArgumentBinder.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/NamedTopicFactories.java
```

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Activated.java
```

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/MapEventTransformerBinding.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Committed.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CoherenceEventListener.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/ElementBinder.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Scope.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/package-info.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Utils.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/AnnotationLiteral.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/SerializerFormat.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Departed.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Removed.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Stopping.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ChainedExtractor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Syncing.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/PropertyExtractor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Started.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/FilterFactory.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Stopped.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Recovered.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Departing.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Error.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ParticipantName.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/exceptions/CoherenceSubscriberException.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/MapEventTransformerFactory.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/FilterBinding.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Synced.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/AnnotatedElementBinder.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/EventArgumentBinderRegistry.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Starting.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ConfigUri.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Inserted.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/SubscriberGroup.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Synchronous.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CommittingRemote.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/ExecutableMethodMapListener.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Assigned.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/FactoryQualifier.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ScopeName.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Committing.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Processor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CommitStrategy.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Connecting.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CoherenceTopicListener.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Executed.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/CoherencePublisherIntroductionAdvice.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Executing.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/SessionName.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CoherencePublisher.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/ExtractorFactories.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CommittingLocal.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/BaseExecutableMethodObserver.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/SubscriberExceptionHandler.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/TypedElementBinder.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Inserting.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/PofExtractor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/ElementValueBinder.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/CoherenceEventListenerProcessor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/AlwaysFilter.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Destroyed.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Deleted.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Rollback.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/AnnotatedMapListener.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Arrived.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Truncated.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/CoherenceTopicListenerProcessor.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ExtractorBinding.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Created.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/ServiceName.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/DefaultTopicBinder.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/View.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Topic.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Replicating.java

* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Updating.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/CacheName.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Name.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Backlog.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/messaging/binders/ElementArgumentBinderRegistry.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Removing.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Activating.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/EventObserverSupport.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/CoherenceFactory.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Lite.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/MapName.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Topics.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/event/ExecutableMethodEventObserver.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Disposing.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Disconnected.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/annotation/Lost.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/NamedCacheFactories.java
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/io/micronaut/coherence/MapEventTransformerFactories.java

No license file was found, but licenses were detected in source scan.

```
#  
# Copyright 2017-2020 original authors  
#  
# Licensed under the Apache License, Version 2.0 (the "License");  
# you may not use this file except in compliance with the License.  
# You may obtain a copy of the License at  
#  
# https://www.apache.org/licenses/LICENSE-2.0  
#  
# Unless required by applicable law or agreed to in writing, software
```



```
# distributed under the License is distributed on an "AS IS" BASIS,  
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
# See the License for the specific language governing permissions and  
# limitations under the License.  
#
```

io.micronaut.coherence.grpc.GrpcServerConfigurer\$GrpcServerConfigurationBean

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/META-  
INF/services/com.oracle.coherence.grpc.proxy.GrpcServerConfiguration  
No license file was found, but licenses were detected in source scan.
```

```
#
```

```
# Copyright 2017-2020 original authors
```

```
#
```

```
# Licensed under the Apache License, Version 2.0 (the "License");
```

```
# you may not use this file except in compliance with the License.
```

```
# You may obtain a copy of the License at
```

```
#
```

```
# https://www.apache.org/licenses/LICENSE-2.0
```

```
#
```

```
# Unless required by applicable law or agreed to in writing, software
```

```
# distributed under the License is distributed on an "AS IS" BASIS,
```

```
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
# See the License for the specific language governing permissions and
```

```
# limitations under the License.
```

```
#
```

io.micronaut.coherence.InjectorImpl

Found in path(s):

```
* /opt/cola/permits/1331473926_1653068590.7919037/0/micronaut-coherence-3-0-1-sources-jar/META-  
INF/services/com.oracle.coherence.inject.Injector
```

1.78 com.helger:profiler 1.1.1

1.78.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<name>The Apache License, Version 2.0</name>
```

```
<url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
```

Found in path(s):

```
* /opt/cola/permits/1293853635_1649177431.36/0/profiler-1-1-1-jar/META-  
INF/maven/com.helger/profiler/pom.xml
```

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.papertrail.profiler
Bnd-LastModified: 1563722217801
Build-Jdk: 1.8.0_202
Built-By: philip
Bundle-Description: A pure-java implementation of the twitter/util project's `CpuProfile` and related classes.
Bundle-DocURL: <http://www.helger.com>
Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>
Bundle-ManifestVersion: 2
Bundle-Name: com.helger:profiler
Bundle-SymbolicName: com.helger.profiler
Bundle-Vendor: Philip Helger
Bundle-Version: 1.1.1
Created-By: Apache Maven Bundle Plugin
Export-Package: com.papertrail.profiler;version="1.1.1",com.papertrail.profiler.jaxrs;uses:="javax.ws.rs";version="1.1.1"
Implementation-Title: com.helger:profiler
Implementation-URL: <https://github.com/phax/profiler>
Implementation-Vendor: Philip Helger
Implementation-Vendor-Id: com.helger
Implementation-Version: 1.1.1
Import-Package: com.papertrail.profiler.javax.ws.rs;version="[2.1,3)"
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-4.1.0.201810181252

Found in path(s):

* /opt/cola/permits/1293853635_1649177431.36/0/profiler-1-1-1-jar/META-INF/MANIFEST.MF

1.79 kafka-protobuf-serializer 5.5.1

1.79.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<parent>
```

```
<groupId>io.confluent</groupId>
```

```
<artifactId>kafka-schema-registry-parent</artifactId>
```

```
<version>5.5.1</version>
```

```
</parent>
```

```
<licenses>
  <license>
    <name>Confluent Community License</name>
    <url>http://www.confluent.io/confluent-community-license</url>
    <distribution>repo</distribution>
  </license>
  <license>
    <name>Apache License 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
    <distribution>repo</distribution>
  </license>
</licenses>
```

```
<artifactId>kafka-protobuf-serializer</artifactId>
<packaging>jar</packaging>
<name>kafka-protobuf-serializer</name>
```

```
<dependencies>
  <dependency>
    <groupId>org.apache.kafka</groupId>
    <artifactId>kafka_${kafka.scala.version}</artifactId>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-protobuf-provider</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java-util</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-serializer</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-client</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry</artifactId>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry</artifactId>
    <type>test-jar</type>
```

```

    <scope>test</scope>
  </dependency>
</dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>connect-api</artifactId>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>kafka-clients</artifactId>
  <classifier>test</classifier>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>kafka_${kafka.scala.version}</artifactId>
  <classifier>test</classifier>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-core</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>

```

```

<build>
  <plugins>
    <plugin>
      <groupId>com.github.os72</groupId>
      <artifactId>protoc-jar-maven-plugin</artifactId>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <version>2.6</version>
      <executions>
        <execution>
          <goals>
            <goal>test-jar</goal>
          </goals>
          <phase>test-compile</phase>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

```
</executions>
</plugin>
</plugins>
</build>
</project>
```

Found in path(s):

* /opt/cola/permits/1337070258_1654207896.050128/0/kafka-protobuf-serializer-5-5-1-jar/META-INF/maven/io.confluent/kafka-protobuf-serializer/pom.xml

1.80 jackson-datatype-jsr310 2.13.2

1.80.1 Available under license :

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.81 jakarta-annotations-api 2.0.0

1.81.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that

are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license

different than this Agreement, provided that such license:

- i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
- ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
- iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
- iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits

and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone

to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source code along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties

who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL

DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands

you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notices for Jakarta Annotations

This content is produced and maintained by the Jakarta Annotations project.

* Project home: <https://projects.eclipse.org/projects/ee4j.ca>

Trademarks

Jakarta Annotations is a trademark of the Eclipse Foundation.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/common-annotations-api>

Third-party Content

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.82 byte-buddy 1.12.7

1.82.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright `${project.inceptionYear}` - `${current.year}` `${copyright.holder}`

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.83 micronaut-oracle-cloud 2.1.1

1.83.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2020 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
*/
```

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1331474080_1653514413.2292626/0/micronaut-oraclecloud-sdk-2-1-1-sources-jar/io/micronaut/oraclecloud/clients/SdkClients.java
- * /opt/cola/permits/1331474080_1653514413.2292626/0/micronaut-oraclecloud-sdk-2-1-1-sources-jar/io/micronaut/oraclecloud/clients/SdkAutomaticFeature.java
- * /opt/cola/permits/1331474080_1653514413.2292626/0/micronaut-oraclecloud-sdk-2-1-1-sources-jar/io/micronaut/oraclecloud/clients/BouncyCastleFeature.java

1.84 kafka-protobuf-provider 5.5.1

1.84.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<parent>
```

```
<groupId>io.confluent</groupId>
```

```
<artifactId>kafka-schema-registry-parent</artifactId>
```

```
<version>5.5.1</version>
```

```
</parent>
```

```
<licenses>
```

```
<license>
```

```
<name>Confluent Community License</name>
```

```
<url>http://www.confluent.io/confluent-community-license</url>
```

```
<distribution>repo</distribution>
```

```
</license>
```

```
<license>
```

```
<name>Apache License 2.0</name>
```

```
<url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
```

```
<distribution>repo</distribution>
```

```
</license>
```

```
</licenses>
```

```
<artifactId>kafka-protobuf-provider</artifactId>
```

```
<packaging>jar</packaging>
```

```
<name>kafka-protobuf-provider</name>
```



```
<dependencies>
  <dependency>
    <groupId>com.squareup.wire</groupId>
    <artifactId>wire-schema</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java-util</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-client</artifactId>
  </dependency>
  <dependency>
    <groupId>org.mockito</groupId>
    <artifactId>mockito-core</artifactId>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>
```

```
<build>
  <plugins>
    <plugin>
      <groupId>com.github.os72</groupId>
      <artifactId>protoc-jar-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
</project>
```

Found in path(s):

* /opt/cola/permits/1337069578_1654207937.429193/0/kafka-protobuf-provider-5-5-1-jar/META-INF/maven/io.confluent/kafka-protobuf-provider/pom.xml

1.85 junit-jupiter-junit-jupiter-engine 5.7.1

1.85.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

3.1 If a Contributor Distributes the Program in any form, then:

* **a)** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **i)** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2**** When the Program is Distributed as Source Code:

* ****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* ****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.86 reactive-relational-database-connectivity-bill-of-materials Arabba-SR11

1.86.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Reactive Relational Database Connectivity

Copyright 2017-2018 the original author or authors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.87 jboss-logging 3.3.2.Final

1.87.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

1.88 feign-httpclient 8.18.0

1.88.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274702072_1645234420.76/0/feign-httpclient-8-18-0-sources-

jar/feign/httpclient/HttpClient.java

1.89 junit-platform-junit-platform-commons

1.8.2

1.89.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of **a)** distributing or **b)** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2**** When the Program is Distributed as Source Code:

* ****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* ****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.90 hamcrest 1.1

1.90.1 Available under license :

BSD License

Copyright (c) 2000-2006, www.hamcrest.org

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY

EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT

SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED

TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR

BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2002-2006 Joe Walnes and QDox Project Team

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.91 caffeine-cache 2.9.1

1.91.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.92 micronaut-reactor 2.2.2

1.92.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2019 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
```

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331473741_1653068550.395305/0/micronaut-reactor-2-2-2-sources-1-jar/io/micronaut/reactor/convert/ReactorConverterRegistrar.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2021 original authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331473741_1653068550.395305/0/micronaut-reactor-2-2-2-sources-1-jar/io/micronaut/reactor/convert/ReactorToRxJava2ConverterRegistrar.java
* /opt/cola/permits/1331473741_1653068550.395305/0/micronaut-reactor-2-2-2-sources-1-jar/io/micronaut/reactor/convert/ReactorToRxJava3ConverterRegistrar.java

1.93 icu4j 70.1

1.93.1 Available under license :

COPYRIGHT AND PERMISSION NOTICE (ICU 58 and later)

Copyright 1991-2020 Unicode, Inc. All rights reserved.

Distributed under the Terms of Use in <https://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use,

copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies of the Data Files or Software, or
(b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

Third-Party Software Licenses

This section contains third-party software notices and/or additional terms for licensed third-party software components included within ICU libraries.

1. ICU License - ICU 1.8.1 to ICU 57.1

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2016 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

2. Chinese/Japanese Word Break Dictionary Data (cjdict.txt)

```
# The Google Chrome software developed by Google is licensed under
# the BSD license. Other software included in this distribution is
# provided under other licenses, as set forth below.
#
# The BSD License
# http://opensource.org/licenses/bsd-license.php
# Copyright (C) 2006-2008, Google Inc.
#
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice,
# this list of conditions and the following disclaimer.
# Redistributions in binary form must reproduce the above
# copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided with
# the distribution.
# Neither the name of Google Inc. nor the names of its
# contributors may be used to endorse or promote products derived from
# this software without specific prior written permission.
#
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
# CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
# INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
```

MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

#

#

The word list in cjdict.txt are generated by combining three word lists
listed below with further processing for compound word breaking. The
frequency is generated with an iterative training against Google web
corpora.

#

* Libtabe (Chinese)

- https://sourceforge.net/project/?group_id=1519

- Its license terms and conditions are shown below.

#

* IPADIC (Japanese)

- <http://chasen.aist-nara.ac.jp/chasen/distribution.html>

- Its license terms and conditions are shown below.

#

-----COPYING.libtabe ---- BEGIN-----

#

/*

* Copyright (c) 1999 TaBE Project.

* Copyright (c) 1999 Pai-Hsiang Hsiao.

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* . Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* . Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the

* distribution.

* . Neither the name of the TaBE Project nor the names of its

* contributors may be used to endorse or promote products derived

* from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

* "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT

* LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS

```
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# /*
# * Copyright (c) 1999 Computer Systems and Communication Lab,
# *      Institute of Information Science, Academia
# *      Sinica. All rights reserved.
# *
# * Redistribution and use in source and binary forms, with or without
# * modification, are permitted provided that the following conditions
# * are met:
# *
# * . Redistributions of source code must retain the above copyright
# * notice, this list of conditions and the following disclaimer.
# * . Redistributions in binary form must reproduce the above copyright
# * notice, this list of conditions and the following disclaimer in
# * the documentation and/or other materials provided with the
# * distribution.
# * . Neither the name of the Computer Systems and Communication Lab
# * nor the names of its contributors may be used to endorse or
# * promote products derived from this software without specific
# * prior written permission.
# *
# * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# * "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# * LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# * FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# * REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
# * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# * (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# * SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# * OF THE POSSIBILITY OF SUCH DAMAGE.
# */
#
# Copyright 1996 Chih-Hao Tsai @ Beckman Institute,
# University of Illinois
# c-tsai4@uiuc.edu http://casper.beckman.uiuc.edu/~c-tsai4
```

```
#
# -----COPYING.libtabe-----END-----
#
#
# -----COPYING.ipadic-----BEGIN-----
#
# Copyright 2000, 2001, 2002, 2003 Nara Institute of Science
# and Technology. All Rights Reserved.
#
# Use, reproduction, and distribution of this software is permitted.
# Any copy of this software, whether in its original form or modified,
# must include both the above copyright notice and the following
# paragraphs.
#
# Nara Institute of Science and Technology (NAIST),
# the copyright holders, disclaims all warranties with regard to this
# software, including all implied warranties of merchantability and
# fitness, in no event shall NAIST be liable for
# any special, indirect or consequential damages or any damages
# whatsoever resulting from loss of use, data or profits, whether in an
# action of contract, negligence or other tortuous action, arising out
# of or in connection with the use or performance of this software.
#
# A large portion of the dictionary entries
# originate from ICOT Free Software. The following conditions for ICOT
# Free Software applies to the current dictionary as well.
#
# Each User may also freely distribute the Program, whether in its
# original form or modified, to any third party or parties, PROVIDED
# that the provisions of Section 3 ("NO WARRANTY") will ALWAYS appear
# on, or be attached to, the Program, which is distributed substantially
# in the same form as set out herein and that such intended
# distribution, if actually made, will neither violate or otherwise
# contravene any of the laws and regulations of the countries having
# jurisdiction over the User or the intended distribution itself.
#
# NO WARRANTY
#
# The program was produced on an experimental basis in the course of the
# research and development conducted during the project and is provided
# to users as so produced on an experimental basis. Accordingly, the
# program is provided without any warranty whatsoever, whether express,
# implied, statutory or otherwise. The term "warranty" used herein
# includes, but is not limited to, any warranty of the quality,
# performance, merchantability and fitness for a particular purpose of
# the program and the nonexistence of any infringement or violation of
# any right of any third party.
#
```

Each user of the program will agree and understand, and be deemed to
have agreed and understood, that there is no warranty whatsoever for
the program and, accordingly, the entire risk arising from or
otherwise connected with the program is assumed by the user.

Therefore, neither ICOT, the copyright holder, or any other
organization that participated in or was otherwise related to the
development of the program and their respective officials, directors,
officers and other employees shall be held liable for any and all
damages, including, without limitation, general, special, incidental
and consequential damages, arising out of or otherwise in connection
with the use or inability to use the program or any product, material
or result produced or otherwise obtained by using the program,
regardless of whether they have been advised of, or otherwise had
knowledge of, the possibility of such damages at any time during the
project or thereafter. Each user will be deemed to have agreed to the
foregoing by his or her commencement of use of the program. The term
"use" as used herein includes, but is not limited to, the use,
modification, copying and distribution of the program and the
production of secondary products from the program.

In the case where the program, whether in its original form or
modified, was distributed or delivered to or received by a user from
any person, organization or entity other than ICOT, unless it makes or
grants independently of ICOT any specific warranty to the user in
writing, such person, organization or entity, will also be exempted
from and not be held liable to the user for any such damages as noted
above as far as the program is concerned.

-----COPYING.ipadic-----END-----

3. Lao Word Break Dictionary Data (laodict.txt)

Copyright (C) 2016 and later: Unicode, Inc. and others.
License & terms of use: <http://www.unicode.org/copyright.html>
Copyright (c) 2015 International Business Machines Corporation
and others. All Rights Reserved.

Project: <https://github.com/rober42539/lao-dictionary>
Dictionary: <https://github.com/rober42539/lao-dictionary/laodict.txt>
License: <https://github.com/rober42539/lao-dictionary/LICENSE.txt>
(copied below)

This file is derived from the above dictionary version of Nov 22, 2020

Copyright (C) 2013 Brian Eugene Wilson, Robert Martin Campbell.
All rights reserved.
#


```
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions are met:
#
# Redistributions of source code must retain the above copyright notice, this
# list of conditions and the following disclaimer. Redistributions in binary
# form must reproduce the above copyright notice, this list of conditions and
# the following disclaimer in the documentation and/or other materials
# provided with the distribution.
#
# THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
# "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
# LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS
# FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE
# COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT,
# INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
# (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR
# SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
# HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
# STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
# ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
# OF THE POSSIBILITY OF SUCH DAMAGE.
# -----
```

4. Burmese Word Break Dictionary Data (burmesedict.txt)

```
# Copyright (c) 2014 International Business Machines Corporation
# and others. All Rights Reserved.
#
# This list is part of a project hosted at:
# github.com/kanyawtech/myanmar-karen-word-lists
#
# -----
# Copyright (c) 2013, LeRoy Benjamin Sharon
# All rights reserved.
#
# Redistribution and use in source and binary forms, with or without
# modification, are permitted provided that the following conditions
# are met: Redistributions of source code must retain the above
# copyright notice, this list of conditions and the following
# disclaimer. Redistributions in binary form must reproduce the
# above copyright notice, this list of conditions and the following
# disclaimer in the documentation and/or other materials provided
# with the distribution.
#
# Neither the name Myanmar Karen Word Lists, nor the names of its
# contributors may be used to endorse or promote products derived
# from this software without specific prior written permission.
#
```

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON
ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR
TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF
THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
SUCH DAMAGE.

5. Time Zone Database

ICU uses the public domain data and code derived from Time Zone Database for its time zone support. The ownership of the TZ database is explained in BCP 175: Procedure for Maintaining the Time Zone Database section 7.

7. Database Ownership

The TZ database itself is not an IETF Contribution or an IETF
document. Rather it is a pre-existing and regularly updated work
that is in the public domain, and is intended to remain in the
public domain. Therefore, BCPs 78 [RFC5378] and 79 [RFC3979] do
not apply to the TZ Database or contributions that individuals make
to it. Should any claims be made and substantiated against the TZ
Database, the organization that is providing the IANA
Considerations defined in this RFC, under the memorandum of
understanding with the IETF, currently ICANN, may act in accordance
with all competent court orders. No ownership claims will be made
by ICANN or the IETF Trust on the database or the code. Any person
making a contribution to the database or code waives all rights to
future claims in that contribution or in the TZ Database.

6. Google double-conversion

Copyright 2006-2011, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above

copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.94 pact-jvm-provider 4.0.10

1.94.1 Available under license :

Apache-2.0

1.95 jcip-annotations-under-apache-license

1.0-1

1.95.1 Available under license :

Apache License, Version 2.0

<http://www.apache.org/licenses/LICENSE-2.0.txt>

1.96 abegocore 1.0.3

1.96.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bnd-LastModified: 1446804256758

Build-Jdk: 1.8.0_60

Built-By: ub

Bundle-Description: Efficient and customizable TreeLayout Algorithm in Java.

Bundle-DocURL: <http://abego-software.de>

Bundle-License: <http://www.abego-software.de/legal/apl-v10.html>

Bundle-ManifestVersion: 2
Bundle-Name: abego TreeLayout Core
Bundle-SymbolicName: org.abego.treelayout.core
Bundle-Vendor: abego Software GmbH, Germany
Bundle-Version: 1.0.3
Created-By: Apache Maven Bundle Plugin
Export-Package: org.abego.treelayout.util;uses:="org.abego.treelayout";version="1.0.3",org.abego.treelayout;version="1.0.3"
Implementation-Title: abego TreeLayout Core
Implementation-Vendor: abego Software GmbH, Germany
Implementation-Vendor-Id: org.abego.treelayout
Implementation-Version: 1.0.3
Import-Package: sun.misc;resolution:=optional
Specification-Title: abego TreeLayout Core
Specification-Vendor: abego Software GmbH, Germany
Specification-Version: 1.0.3
Tool: Bnd-1.50.0

Found in path(s):

* /opt/cola/permits/1162863034_1649022203.09/0/org-abego-treelayout-core-1-0-3-jar/META-INF/MANIFEST.MF

1.97 zstd-jni 1.5.0-2

1.97.1 Available under license :

BSD License

For Zstandard software

Copyright (c) 2016-present, Facebook, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name Facebook nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED

WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Zstd-jni: JNI bindings to Zstd Library

Copyright (c) 2015-present, Luben Karavelov/ All rights reserved.

BSD License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This

General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains

a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but

does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to

control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then

the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals

of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

Apache License

Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.98 apache-http-client 5.0.3

1.98.1 Available under license :

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

=====

This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <<http://mozilla.org/MPL/2.0/>>

Mozilla Public License Version 2.0

=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and

all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark)

Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and

(b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

(a) for any code that a Contributor has removed from Covered Software; or

(b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or

(c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its

Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the

Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such

Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
*
*
* 6. Disclaimer of Warranty
* -----
*
* Covered Software is provided under this License on an "as is"
* basis, without warranty of any kind, either expressed, implied, or
* statutory, including, without limitation, warranties that the
* Covered Software is free of defects, merchantable, fit for a
* particular purpose or non-infringing. The entire risk as to the
* quality and performance of the Covered Software is with You.
* Should any Covered Software prove defective in any respect, You
* (not any Contributor) assume the cost of any necessary servicing,
* repair, or correction. This disclaimer of warranty constitutes an
* essential part of this License. No use of any Covered Software is
* authorized under this License except under this disclaimer.
*
```

```
*
*
* 7. Limitation of Liability
* -----
*
* Under no circumstances and under no legal theory, whether tort
* (including negligence), contract, or otherwise, shall any
```

* Contributor, or anyone who distributes Covered Software as *
 * permitted above, be liable to You for any direct, indirect, *
 * special, incidental, or consequential damages of any character *
 * including, without limitation, damages for lost profits, loss of *
 * goodwill, work stoppage, computer failure or malfunction, or any *
 * and all other commercial damages or losses, even if such party *
 * shall have been informed of the possibility of such damages. This *
 * limitation of liability shall not apply to liability for death or *
 * personal injury resulting from such party's negligence to the *
 * extent applicable law prohibits such limitation. Some *
 * jurisdictions do not allow the exclusion or limitation of *
 * incidental or consequential damages, so this exclusion and *
 * limitation may not apply to You. *

*

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.99 opentracing-noop 0.33.0

1.99.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright 2016-2019 The OpenTracing Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/META-INF/maven/io.opentracing/opentracing-noop/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016-2019 The OpenTracing Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopSpan.java

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopSpanContext.java

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopTracerFactory.java

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopTracer.java

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopScopeManager.java

* /opt/cola/permits/1340188505_1654650482.1626384/0/opentracing-noop-0-33-0-sources-2-jar/io/opentracing/noop/NoopSpanBuilder.java

1.100 javax-ws-rs-api 2.1.1

1.100.1 Available under license :

Found license 'General Public License 2.0' in '* Copyright (c) 2013, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2010, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in '* Copyright (c) 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2012, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License, * This method is reserved for future use. Proprietary JAX-RS extensions may leverage the method.'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2012, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2015, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2011, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2013, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in '* Copyright (c) 2012, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License, * This method is reserved for future use. Proprietary JAX-RS extensions may leverage the method.'

Found license 'General Public License 2.0' in '* Copyright (c) 2011, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in '* Copyright (c) 2010, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in '* Copyright (c) 2012, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in '* Copyright (c) 2015, 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v. 2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in '* Copyright (c) 2017 Oracle and/or its affiliates. All rights reserved. * This program and the accompanying materials are made available under the * terms of the Eclipse Public License v.

2.0, which is available at * Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

1.101 micrometer-bom 1.8.3

1.101.1 Available under license :

This product includes 'ANTLR', which is released under the following license(s):

BSD-3 <<http://opensource.org/licenses/BSD-3-Clause>>

This product includes 'ASM', which is released under the following license(s):

ASM (BSD-3) <<http://asm.ow2.org/license.html>>

This product includes 'Apache Cassandra', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Apache Commons CLI', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Apache Commons Codec', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Apache Commons Logging', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'GSON', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Guava', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'HttpClient', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'JAXB', which is released under the following license(s):

CDDL-1.0 <<https://opensource.org/licenses/CDDL-1.0>>

This product includes 'JRegex', which is released under the following license(s):

BSD-3 <<http://opensource.org/licenses/BSD-3-Clause>>

This product includes 'JSON.simple', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Jackson', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'Javassist', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

This product includes 'LogBack', which is released under the following license(s):

Eclipse Public License 1.0 <<http://www.eclipse.org/legal/epl-v10.html>>

This product includes 'Reflections', which is released under the following license(s):

Do What The Fuck You Want To Public License (WTFPL) <<http://www.wtfpl.net>>

This product includes 'SLF4J', which is released under the following license(s):

The MIT License (MIT) <<http://opensource.org/licenses/MIT>>

This product includes 'SnakeYAML', which is released under the following license(s):

Apache License 2.0 <<http://www.apache.org/licenses/LICENSE-2.0>>

All other components of this product are: Copyright (c) 2010-2017 New Relic, Inc. All rights reserved.

Certain inventions disclosed in this file may be claimed within patents owned or patent applications filed by New Relic, Inc. or third parties. Subject to the terms of this notice, New Relic grants you a nonexclusive, nontransferable license, without the right to sublicense, to (a) install and execute one copy of these files on any number of workstations owned or controlled by you and (b) distribute verbatim copies of these files to third parties. As a condition to the foregoing grant, you must provide this notice along with each copy you distribute and you must not remove, alter, or obscure this notice.

All other use, reproduction, modification, distribution, or other exploitation of these files is strictly prohibited, except as may be set forth in a separate written license agreement between you and New Relic. The terms of any such license agreement will control over this notice. The license stated above will be automatically terminated and revoked if you exceed its scope or violate any of the terms of this notice.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of New Relic, except as required for reasonable and customary use in describing the origin of this file and reproducing the content of this notice. You may not mark or brand this file with any trade name, trademarks, service marks, or product names other than the original brand (if any) provided by New Relic.

Unless otherwise expressly agreed by New Relic in a separate written license agreement, these files are provided AS IS, WITHOUT WARRANTY OF ANY KIND, including without any implied warranties of MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, or NON-INFRINGEMENT. As a condition to your use of these files, you are solely responsible for such use. New Relic will have no liability to you for direct, indirect, consequential, incidental, special, or punitive damages or for lost profits or data.

Micrometer

Copyright (c) 2017-Present VMware, Inc. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'io.netty.util.internal.logging',
in the Netty/Common library distributed by The Netty Project:

- * Copyright 2013 The Netty Project
- * License: Apache License v2.0
- * Homepage: <https://netty.io>

This product contains a modified portion of 'StringUtils.isBlank()',
in the Commons Lang library distributed by The Apache Software Foundation:

- * Copyright 2001-2019 The Apache Software Foundation
- * License: Apache License v2.0
- * Homepage: <https://commons.apache.org/proper/commons-lang/>

This product contains a modified portion of 'JsonUtf8Writer',
in the Moshi library distributed by Square, Inc:

- * Copyright 2010 Google Inc.
- * License: Apache License v2.0
- * Homepage: <https://github.com/square/moshi>

This product contains a modified portion of the 'org.springframework.lang'
package in the Spring Framework library, distributed by VMware, Inc:

- * Copyright 2002-2019 the original author or authors.
- * License: Apache License v2.0
- * Homepage: <https://spring.io/projects/spring-framework>

All components of this product are: Copyright (c) 2010-2015 New Relic, Inc. All rights reserved.

Certain inventions disclosed in this file may be claimed within patents owned or patent applications filed by New Relic, Inc. or third parties. Subject to the terms of this notice, New Relic grants you a nonexclusive, nontransferable license, without the right to sublicense, to (a) install and execute one copy of these files on any number of workstations owned or controlled by you and (b) distribute verbatim copies of these files to third parties. As a condition to the foregoing grant, you must provide this notice along with each copy you distribute and you must not remove, alter, or obscure this notice.

All other use, reproduction, modification, distribution, or other exploitation of these files is strictly prohibited, except as may be set forth in a separate written license agreement between you and New Relic. The terms of any such license agreement will control over this notice. The license stated above will be automatically terminated and revoked if you exceed its scope or violate any of the terms of this notice.

This License does not grant permission to use the trade names, trademarks, service marks, or product names of New Relic, except as required for reasonable and customary use in describing the origin of this file and reproducing the content of this notice. You may not mark or brand this file with any trade name, trademarks, service marks, or product names other than the original brand (if any) provided

by New Relic.

Unless otherwise expressly agreed by New Relic in a separate written license agreement, these files are provided AS IS, WITHOUT WARRANTY OF ANY KIND, including without any implied warranties of MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, or NON-INFRINGEMENT. As a condition to your use of these files, you are solely responsible for such use. New Relic will have no liability to you for direct, indirect, consequential, incidental, special, or punitive damages or for lost profits or data.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.102 apache-commons-collections 4.1

1.102.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Collections

Copyright 2001-2015 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

1.103 metrics-utility-servlets 4.1.17

1.103.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.codahale.metrics.servlets

Build-Jdk-Spec: 1.8

Bundle-Description: A set of utility servlets for Metrics, allowing you to expose valuable information about your production environment.

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>

Bundle-ManifestVersion: 2

Bundle-Name: Metrics Utility Servlets

Bundle-SymbolicName: io.dropwizard.metrics.servlets

Bundle-Version: 4.1.17

Created-By: Apache Maven Bundle Plugin

Export-Package: com.codahale.metrics.servlets;uses:="com.codahale.metrics,com.codahale.metrics.health,com.fasterxml.jackson.databind,javax.servlet,javax.servlet.http";version="4.1.17"

Implementation-Title: Metrics Utility Servlets

Implementation-Version: 4.1.17

Import-Package: com.codahale.metrics;version="[4.1,5)",com.codahale.metrics.health;version="[4.1,5)",com.codahale.metrics.json;version="[4.1,5)",com.codahale.metrics.jvm;version="[4.1,5)",com.fasterxml.jackson.databind;version="[2.9,3)",com.fasterxml.jackson.databind.util;version="[2.9,3)",com.papertrail.profiler;version="[1.1,2)",javax.servlet;version="[2.5.0,4.0.0)",javax.servlet.http;version="[2.5.0,4.0.0)"

Private-Package: com.codahale.metrics.servlets

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"

Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188508_1654734043.7976656/0/metrics-servlets-4-1-17-jar/META-INF/MANIFEST.MF

1.104 java-diff-utils 1.3.0

1.104.1 Available under license :

No license file was found, but licenses were detected in source scan.

<url><http://www.apache.org/licenses/LICENSE-2.0.txt></url>

Found in path(s):

* /opt/cola/permits/1131003422_1649015566.9/0/diffutils-1-3-0-jar/META-INF/maven/com.googlecode.java-diff-utils/diffutils/pom.xml

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Archiver-Version: Plexus Archiver

Built-By: smitterm

Created-By: Apache Maven Bundle Plugin

Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.txt>

Bnd-LastModified: 1362404972142

Export-Package: difflib;uses:="difflib.myers";version="1.3.0",difflib.myers;uses:="difflib;version="1.3.0"

Bundle-Version: 1.3.0
Bundle-Description: The DiffUtils library for computing diffs, applying patches, generating side-by-side view in Java.
Bundle-Name: java-diff-utils
Build-Jdk: 1.5.0_22
Bundle-ManifestVersion: 2
Bundle-SymbolicName: com.googlecode.java-diff-utils.diffutils
Tool: Bnd-1.50.0

Found in path(s):

* /opt/cola/permits/1131003422_1649015566.9/0/diffutils-1-3-0-jar/META-INF/MANIFEST.MF

1.105 iterators 1.5

1.105.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017 dmfs GmbH  
*  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/utils/SlimSet.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/FilteredIterator.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/SerialIterator.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/UnquotedSplit.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/Filter.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/decorators/Flattened.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/filters/NoneOf.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/SerializableIterator.java
* /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/decorators/Reverse.java

- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/Repeatable.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/Distinct.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/filters/NonNull.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/decorators/Flattened.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/Function.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/filters/Skip.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/decorators/Serialized.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/CachingIterable.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/AbstractBaseIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/AbstractFilteredIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/DistinctIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/CsvIterable.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/EmptyIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/SingletonIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/filters/Distinct.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/CsvIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/decorators/Mapped.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/decorators/Fluent.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/decorators/Filtered.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/filters/AnyOf.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/decorators/Mapped.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/decorators/Filtered.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/ArrayIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/Split.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/FluentIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/AbstractConvertedIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/Split.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/SerialIteratorIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/ConvertedIterator.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterators/UnquotedSplit.java
- * /opt/cola/permits/1162864298_1620923965.62/0/iterators-1-5-sources-1-jar/org/dmfs/iterables/ArrayIterable.java

1.106 atomicfu-common 0.14.1

1.106.1 Available under license :

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                    ==
== in this case for the kotlix.atomicfu library.        ==
=====
```

kotlix.atomcfu library.

Copyright 2017 JetBrains s.r.o and respective authors and developers

/*

* Copyright 2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

1.107 io-swagger-swagger-annotations 1.6.0

1.107.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bnd-LastModified: 1573911985726

Build-Jdk: 1.8.0_151

Built-By: frantuma

Bundle-Description: Sonatype helps open source projects to set up Maven repositories on <https://oss.sonatype.org/>

Bundle-License: <http://www.apache.org/licenses/LICENSE-2.0.html>

Bundle-ManifestVersion: 2

Bundle-Name: swagger-annotations

Bundle-SymbolicName: io.swagger.annotations

Bundle-Version: 1.6.0

Created-By: Apache Maven Bundle Plugin

Export-Package: io.swagger.annotations;version="1.6.0"

implementation-version: 1.6.0
mode: development
package: io.swagger
Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.3.0.201609221906
url: <https://github.com/swagger-api/swagger-core/modules/swagger-annotations>

Found in path(s):

* /opt/cola/permits/1337070078_1654214889.0428526/0/swagger-annotations-1-6-0-jar/META-INF/MANIFEST.MF

1.108 error_prone_annotations 2.7.1

1.108.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2014 The Error Prone Authors.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/NoAllocation.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/LockMethod.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/UnlockMethod.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017 The Error Prone Authors.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.
```


* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/OverridingMethodsMustInvokeSuper.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/DoNotCall.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/concurrent/GuardedBy.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CheckReturnValue.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Error Prone Authors.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/FormatString.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/CompatibleWith.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/RestrictedApi.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/MustBeClosed.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/DoNotMock.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/FormatMethod.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/Var.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/concurrent/LazyInit.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/IncompatibleModifiers.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/RequiredModifiers.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/Immutable.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/CompileTimeConstant.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/SuppressPackageLocation.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/ForOverride.java

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-

jar/com/google/errorprone/annotations/CanIgnoreReturnValue.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2021 The Error Prone Authors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMeValidationDisabled.java
* /opt/cola/permits/1288519880_1647351834.56/0/error-prone-annotations-2-7-1-sources-jar/com/google/errorprone/annotations/InlineMe.java

1.109 servicelocator-default-implementation

2.6.1

1.109.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which

are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall

apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness

for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged

intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of

the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),

version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis

or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other

recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not

signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: <https://projects.eclipse.org/projects/ee4j.glassfish>

Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For

more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/glassfish-ha-api>
- * <https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor>
- * <https://github.com/eclipse-ee4j/glassfish-shoal>
- * <https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck>
- * <https://github.com/eclipse-ee4j/glassfish-jsftemplating>
- * <https://github.com/eclipse-ee4j/glassfish-hk2-extra>
- * <https://github.com/eclipse-ee4j/glassfish-hk2>
- * <https://github.com/eclipse-ee4j/glassfish-fighterfish>

Third-party Content

This project leverages the following third party content.

None

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.110 jackson-jaxrs 2.13.2

1.110.1 Available under license :

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>
Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.
It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

1.111 arrow-type-classes 0.9.0

1.111.1 Available under license :

Copyright (C) 2017 The Arrow Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

A copy of Apache License 2.0 has been included at the end of this file.

Code in Arrow is derived in part from typelevel/Cats, Shapeless and Scalaz.

Cats license follows:

Cats Copyright (c) 2015 Erik Osheim.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Scalaz license follows:

Copyright (c) 2009-2014 Tony Morris, Runar Bjarnason, Tom Adams, Kristian Domagala, Brad Clow, Ricky Clarkson, Paul Chiusano, Trygve Laugstøl, Nick Partridge, Jason Zaugg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code in Arrow Recursion is derived in part from aedans/Katalyst.

Katalyst license follows:

MIT License

Copyright (c) 2017 Aedan Smith

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Code in Arrow Optics is derived in part from julien-truffaut/Monocle.

Monocle license follows:

The MIT License (MIT)

Copyright (c) 2014 Julien Truffaut

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Duration.kt license follows:

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [2016] [sksamuel]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

Code in Arrow is derived in part from frees-io/freestyle.

Freestyle license follows:

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (C) 2017 47 Degrees. <<http://47deg.com>>

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Code in Arrow is derived in part from h0tk3y/probabilistik.

probabilistik license follows:

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Code in Arrow is derived in part from functional-streams-for-scala/fs2.

fs2 license follows:

The MIT License (MIT)

Copyright (c) 2013 Paul Chiusano, and respective contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Code in FS2 is derived in part from Cats. The Cats license is as follows:

Cats Copyright (c) 2015 Erik Osheim.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Code in Cats is derived in part from Scalaz. The Scalaz license follows:

Copyright (c) 2009-2014 Tony Morris, Runar Bjarnason, Tom Adams, Kristian Domagala, Brad Clow, Ricky Clarkson, Paul Chiusano, Trygve Laugstøl, Nick Partridge, Jason Zaugg. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code in Arrow is derived in part from <https://github.com/monix/monix>

Monix license follows:

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Code in Arrow is derived in part from <https://github.com/akka/akka>

Akka license follows:

This software is licensed under the Apache 2 license, quoted below.

Copyright 2009-2018 Lightbend Inc. <<https://www.lightbend.com>>

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.112 atomicfu 0.14.1

1.112.1 Available under license :

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                      ==
== in this case for the kotlix.atomicfu library.         ==
=====
```

kotlix.atomcfu library.

Copyright 2017 JetBrains s.r.o and respective authors and developers

/*

* Copyright 2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

1.113 io-grpc-grpc-context 1.39.0

1.113.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified portion of 'OkHttp', an open source
HTTP & SPDY client for Android and Java applications, which can be obtained
at:

* LICENSE:
* okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
* HOMEPAGE:
* <https://github.com/square/okhttp>
* LOCATION_IN_GRPC:
* okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source
cloud-native high-performance edge/middle/service proxy, which can be
obtained at:

* LICENSE:
* xds/third_party/envoy/LICENSE (Apache License 2.0)
* NOTICE:
* xds/third_party/envoy/NOTICE
* HOMEPAGE:

- * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
- * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
- * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:
- * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
- * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
- * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa', an open source universal data plane API, which can be obtained at:

- * LICENSE:
- * xds/third_party/udpa/LICENSE (Apache License 2.0)
- * HOMEPAGE:
- * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
- * xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.114 hibernate-validator 6.1.7.Final

1.114.1 Available under license :

Hibernate Validator, declare and validate application constraints

License: Apache License, Version 2.0

See the license.txt file in the root directory or <<http://www.apache.org/licenses/LICENSE-2.0>>.

Adam Stawicki

Ahmed Al Hafoudh

Alaa Nassef

Andrey Derevyanko
Andrey Rodionov
Asutosh Pandya
Benson Margulies
Brent Douglas
Carlos Vara
Carlo de Wolf
Chris Beckey
Christian Ivan
Dag Hovland
Damir Alibegovic
Dario Seidl
Davide D'Alto
Davide Marchignoli
Denis Tiago
Doug Lea
Emmanuel Bernard
Efthymis Sarbanis
Federico
Federico Mancini
Gavin King
George Gastaldi
Gerhard Petracek
Guillaume Husta
Guillaume Smet
Gunnar Morling
Hardy Ferentschik
Henno Vermeulen
Hillmer Chona
Jan-Willem Willebrands
Jason T. Greene
Jesper Preuss
Jiri Bilek
Julien Furgerot
Julien May
Juraci Krohling
Justin Nauman
Kathryn Killebrew
Kazuki Shimizu
Kevin Pollet
Khalid Alqinyah
Lee KyoungIl
Leonardo Loch Zanivan
Lucas Pouzac
Lukas Niemeier
Mark Hobson
Marko Bekhta
Matthias Kurz

Mert Caliskan
Michal Fotyga
Nicola Ferraro
Nicolas Franois
Paolo Perrotta
Pete Muir
Rob Dickinson
Sanne Grinovero
Sebastian Bayerl
Shahram Goodarzi
Shane Bryzak
Shelly McGowan
Sjaak Derksen
Steve Ebersole
Strong Liu
Tadhg Pearson
Takashi Aoe
Tomaz Cerar
Tommy Johansen
Victor Rezende dos Santos
Willi Schnborn
Xavier Sosnovsky
Yanming Zhou
Yoann Rodire

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity

exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided

that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity,

or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.115 pact-jvm-consumer-junit5 4.0.10

1.115.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.116 joda-time v2.10.9

1.116.1 Available under license :

=====

= NOTICE file corresponding to section 4d of the Apache License Version 2.0 =

=====

This product includes software developed by
Joda.org (<https://www.joda.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.117 okio 2.5.0

1.117.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2019 Square, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Buffer.kt  
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Timeout.kt  
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/BufferedSink.kt
```

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/RealBufferedSource.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/RealBufferedSink.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Sink.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Okio.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/BufferedSource.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Source.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/internal/Buffer.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2016 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Options.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Pipe.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Throttler.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/-
DeprecatedOkio.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-
jar/commonMain/okio/PeekSource.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/-
DeprecatedUtf8.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/-
DeprecatedUpgrade.kt

No license file was found, but licenses were detected in source scan.

/*

- * Copyright (C) 2014 Square, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Source.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-
jar/jvmMain/okio/RealBufferedSink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-
jar/jvmMain/okio/ForwardingSource.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/DeflaterSink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/BufferedSource.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/GzipSink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/ForwardingSink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/BufferedSink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/GzipSource.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/InflaterSource.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/JvmOkio.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Sink.kt
- * /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-

jar/commonMain/okio/SegmentPool.kt

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Segment.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Timeout.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/Buffer.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/AsyncTimeout.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/RealBufferedSource.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/internal/SegmentedByteString.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/internal/RealBufferedSink.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/internal/RealBufferedSource.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/Utf8.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/SegmentedByteString.kt

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/ForwardingTimeout.kt

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/SegmentedByteString.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 Square Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/ByteString.kt
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-
jar/commonMain/okio/internal/ByteString.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/-Platform.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/internal/-
Utf8.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/commonMain/okio/ByteString.kt
* /opt/cola/permits/1337069350_1654119432.2092295/0/okio-2-5-0-sources-5-jar/jvmMain/okio/-Platform.kt

1.118 fabric8-:::-kubernetes-model-:::- certificates 4.10.3

1.118.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object

form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a

file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.119 protocol-buffer-java-util-package 3.17.2

1.119.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.google.protobuf.util
Bnd-LastModified: 1622656380159
Build-Jdk: 1.8.0_181-google-v7
Built-By: deannagarcia
Bundle-Description: Utilities for Protocol Buffers
Bundle-DocURL: <https://developers.google.com/protocol-buffers/>
Bundle-License: <https://opensource.org/licenses/BSD-3-Clause>
Bundle-ManifestVersion: 2
Bundle-Name: Protocol Buffers [Util]
Bundle-SymbolicName: com.google.protobuf.util
Bundle-Version: 3.17.2
Created-By: Apache Maven Bundle Plugin
Export-Package: com.google.protobuf.util;version="3.17.2";uses:="com.google.protobuf"
Import-Package: com.google.common.base;version="[30.1,31)",com.google.common.io;version="[30.1,31)",com.google.common.math;version="[30.1,31)",com.google.common.primitives;version="[30.1,31)",com.google.gson;version="[2.8,3)",com.google.gson.stream;version="[2.8,3)",com.google.protobuf;version="[3.17,4)"
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.7))"
Tool: Bnd-3.0.0.201509101326

Found in path(s):

1.120 jetbrains-kotlin-kotlin-stdlib-jdk8 1.3.71

1.120.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2017 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1162863374_1649200710.32/0/kotlin-stdlib-jdk8-1-3-71-sources-
jar/kotlin/collections/Collections.kt
* /opt/cola/permits/1162863374_1649200710.32/0/kotlin-stdlib-jdk8-1-3-71-sources-
jar/kotlin/internal/jdk8/JDK8PlatformImplementations.kt
* /opt/cola/permits/1162863374_1649200710.32/0/kotlin-stdlib-jdk8-1-3-71-sources-jar/kotlin/streams/Streams.kt
* /opt/cola/permits/1162863374_1649200710.32/0/kotlin-stdlib-jdk8-1-3-71-sources-
jar/kotlin/text/RegexExtensions.kt
```

1.121 opentest4j-opentest4j 1.2.0

1.121.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.122 fabric8-:::-kubernetes-model-:::- scheduling 4.10.3

1.122.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.123 apache-groovy 3.0.9

1.123.1 Available under license :

ASM License

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS

INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Groovy

Copyright 2003-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

This product includes/uses ANTLR2 (<http://www.antlr2.org/>)
developed by Terence Parr 1989-2006

This product includes/uses ANTLR4 (<https://github.com/antlr/antlr4>)
Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

ANTLR 2 License

Antlr2 is released in the public domain.
See licenses/antlr2-license.txt for details.

ANTLR 4 License

Antlr4 is released under a BSD 3-clause license.
See licenses/antlr4-license.txt for details.

ASM License

ASM uses a 3-clause BSD license. For details, see licenses/asm-license.txt.
ANTLR 2 License

We reserve no legal rights to the ANTLR--it is fully in the public domain. An individual or company may do

whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs (commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

In countries where the Public Domain status of the work may not be valid, the author grants a copyright licence to the general public to deal in the work without restriction and permission to sublicense derivatives under the terms of any (OSI approved) Open Source licence.

The Python parser generator code under `antlr/actions/python/` is covered by the 3-clause BSD licence (this part is included in the binary JAR files); the run-time part under `lib/python/` is covered by the GNU GPL, version 3 or later (this part is not included in the binary JAR files). See [1] for the full details.

<https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=750643#80%22>
ANTLR 4 License

[The "BSD 3-clause license"]

Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.124 kotson 2.5.0

1.124.1 Available under license :

MIT

1.125 ktor-server-core 1.2.6

1.125.1 Available under license :

/*

* Copyright 2014-2019 JetBrains s.r.o and contributors. Use of this source code is governed by the Apache 2.0 license.

*/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.126 commons-io 2.11.0

1.126.1 Available under license :

Apache Commons IO
Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.127 mockito-junit-jupiter 4.3.1

1.127.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.128 jersey-inject-hk2 2.32

1.128.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at

<http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/>
Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations,

interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license

before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in

Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335

USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must

be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively

when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the

program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not

excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by

the Free Software Foundation; either version 2 of the License, or
(at your option) any later version.

This program is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this
when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type  
'show w'. This is free software, and you are welcome to redistribute  
it under certain conditions; type 'show c' for details.
```

The hypothetical commands `'show w'` and `'show c'` should show the
appropriate parts of the General Public License. Of course, the commands
you use may be called something other than `'show w'` and `'show c'`; they
could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the program, if
necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the  
program 'Gnomovision' (which makes passes at compilers) written by  
James Hacker.
```

```
signature of Ty Coon, 1 April 1989  
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications
with the library. If this is what you want to do, use the GNU Library
General Public License instead of this License.

```
## CLASSPATH EXCEPTION
```

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.129 reflections 0.9.10

1.129.1 Available under license :

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
Version 2, December 2004

Copyright (C) 2004 Sam Hocevar <sam@hocevar.net>

Everyone is permitted to copy and distribute verbatim or modified copies of this license document, and changing it is allowed as long as the name is changed.

DO WHAT THE FUCK YOU WANT TO PUBLIC LICENSE
TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. You just DO WHAT THE FUCK YOU WANT TO.

1.130 jersey-ext-bean-validation 2.32

1.130.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>

* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

* License: Apache License, 2.0

* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

* License: Apache License, 2.0

* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

* License: Apache License, 2.0

* Project: <http://www.javassist.org/>

* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

* License: jquery.org/license

* Project: jquery.org

* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

* License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)

* Project: <http://www.pasella.it/projects/jquery/barcode>

* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

* License: CC0

* No copyright

* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

* License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)

* Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>

* Copyright: Eric Rowell

org.objectweb.asm Version 8.0

* License: Modified BSD (<http://asm.objectweb.org/license.html>)

* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

* License: Apache License, 2.0

* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other

form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the

patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any

party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance

claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free

software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

- b) Accompany it with a written offer, valid for at least three

years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the

original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to

attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program

into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notice for Jersey Bean Validation module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0

- * Project: <https://beanvalidation.org/>

- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

1.131 asm-commons 8.0.1

1.131.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new { @link GeneratorAdapter}. <i>Subclasses must not use this constructor</i>.
 * Instead, they must use the { @link #GeneratorAdapter(int, MethodVisitor, int, String, String)}
 * version.
 *
 * @param methodVisitor the method visitor to which this adapter delegates calls.
 * @param access the method's access flags (see { @link Opcodes}).
 * @param name the method's name.
 * @param descriptor the method's descriptor (see { @link Type}).
 * @throws IllegalStateException if a subclass calls this constructor.
 */
```

Found in path(s):

- * `/opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-jar/org/objectweb/asm/commons/GeneratorAdapter.java`

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions  
// are met:  
// 1. Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// 2. Redistributions in binary form must reproduce the above copyright  
// notice, this list of conditions and the following disclaimer in the  
// documentation and/or other materials provided with the distribution.  
// 3. Neither the name of the copyright holders nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/ModuleTargetAttribute.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/Remapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/InstructionAdapter.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/JSRInlinerAdapter.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/AnnotationRemapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/StaticInitMerger.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/ModuleResolutionAttribute.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/AnalyzerAdapter.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/RecordComponentRemapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/SimpleRemapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/SignatureRemapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/AdviceAdapter.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/Method.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/CodeSizeEvaluator.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/TryCatchBlockSorter.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-  
jar/org/objectweb/asm/commons/ModuleRemapper.java  
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
```

jar/org/objectweb/asm/commons/MethodRemapper.java
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/ClassRemapper.java
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/ModuleHashesAttribute.java
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/TableSwitchGenerator.java
* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/FieldRemapper.java
No license file was found, but licenses were detected in source scan.

```
// All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions  
// are met:  
// 1. Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// 2. Redistributions in binary form must reproduce the above copyright  
// notice, this list of conditions and the following disclaimer in the  
// documentation and/or other materials provided with the distribution.  
// 3. Neither the name of the copyright holders nor the names of its  
// this software without specific prior written permission.  
/**  
 * Constructs a new { @link LocalVariablesSorter}. <i>Subclasses must not use this constructor</i>.  
 * Instead, they must use the { @link #LocalVariablesSorter(int, int, String, MethodVisitor)}  
 * version.  
 *  
 * @param access access flags of the adapted method.  
 * @param descriptor the method's descriptor (see { @link Type}).  
 * @param methodVisitor the method visitor to which this adapter delegates calls.  
 * @throws IllegalStateException if a subclass calls this constructor.  
 */
```

Found in path(s):

* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/LocalVariablesSorter.java
No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

```
* All rights reserved.  
*  
* Redistribution and use in source and binary forms, with or without  
* modification, are permitted provided that the following conditions  
* are met:  
* 1. Redistributions of source code must retain the above copyright  
* notice, this list of conditions and the following disclaimer.  
* 2. Redistributions in binary form must reproduce the above copyright  
* notice, this list of conditions and the following disclaimer in the
```

* documentation and/or other materials provided with the distribution.
* 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/package.html

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions  
// are met:  
// 1. Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// 2. Redistributions in binary form must reproduce the above copyright  
// notice, this list of conditions and the following disclaimer in the  
// documentation and/or other materials provided with the distribution.  
// 3. Neither the name of the copyright holders nor the names of its  
// this software without specific prior written permission.  
/**  
 * Constructs a new {@link SerialVersionUIDAdder}. <i>Subclasses must not use this  
 * constructor</i>. Instead, they must use the {@link #SerialVersionUIDAdder(int, ClassVisitor)}  
 * version.  
 *  
 * @param classVisitor a {@link ClassVisitor} to which this visitor will delegate calls.  
 * @throws IllegalStateException If a subclass calls this constructor.  
 */
```

Found in path(s):

* /opt/cola/permits/1104664946_1604644137.35/0/asm-commons-8-0-1-sources-1-
jar/org/objectweb/asm/commons/SerialVersionUIDAdder.java

1.132 evo-inflector 1.2.2

1.132.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2013 Atteo

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.133 archaius-core 0.7.6

1.133.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices

that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be

enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.134 micronaut-security 3.5.0

1.134.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2022 original authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/X509AuthenticationArgumentBinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/endpoints/ControllerConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/rules/SensitiveEndpointRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/X509AuthenticationFetcher.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/X509ConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/token/propagation/AbstractOutgoingRequestProcessorMatcher.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/X509Configuration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/utis/LoggingUtils.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/X509Authentication.java

No license file was found, but licenses were detected in source scan.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):

* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/META-INF/native-image/io.micronaut.security/micronaut-security/native-image.properties

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2021 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*

* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/filters/SecurityFilterConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/ServerAuthentication.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/AbstractPrincipalArgumentBinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/filters/SecurityFilterConfigurationProperties.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2020 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/AuthenticationRequest.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/endpoints/introspection/IntrospectionRequest.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/token/validator/TokenValidator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/handlers/LoginHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/BasicAuthUtils.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-

jar/io/micronaut/security/authentication/AuthenticationMode.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/filters/AuthenticationFetcher.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/generator/TokenGenerator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/ConfigurationInterceptUrlMapRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/SecurityRuleResult.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/LoginController.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/TokenAuthenticationFetcher.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/AbstractSecurityRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/generator/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/reader/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/validator/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/handlers/RedirectingLoginHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/UsernamePasswordCredentials.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/event/LoginFailedEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/handlers/LogoutHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/IpPatternsRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/DefaultIntrospectionProcessor.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationResponse.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/event/RefreshTokenGeneratedEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/SecurityConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/config/TokenConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/RolesFinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-

jar/io/micronaut/security/event/LogoutEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/PrincipalArgumentBinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/reader/HttpHeaderTokenReader.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/IssuingAnAccessTokenErrorCode.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/HttpHeaderTokenPropagator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/handlers/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/HttpHeaderTokenPropagatorConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/SecuredAnnotationRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/IntrospectionProcessor.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/Authenticator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/utis/SecurityService.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationModeCondition.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/utis/DefaultSecurityService.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/LogoutControllerConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/PriorToLoginPersistence.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/event/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/LogoutController.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/event/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/BasicAuthAuthenticationConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/config/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/TokenPropagator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-

jar/io/micronaut/security/token/refresh/RefreshTokenPersistence.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/ErrorCode.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/IntrospectionConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/Authentication.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/AuthenticationStrategy.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/reader/TokenResolver.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/reader/DefaultTokenResolver.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/LogoutControllerConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/TokenAuthenticationFactory.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/IntrospectionConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/BasicAuthAuthenticationFetcher.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/generator/RefreshTokenGenerator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/filters/SecurityFilter.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/config/TokenConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/SecurityRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/reader/TokenReader.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/Claims.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/filters/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/TokenCookieConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationFailed.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/jackson/SecurityJacksonModule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/DefaultAuthorizationExceptionHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/SecurityConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/ObtainingAuthorizationErrorCode.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-

jar/io/micronaut/security/endpoints/introspection/IntrospectionController.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/endpoints/introspection/IntrospectionResponse.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/RefreshRedirectConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/HttpHeaderTokenPropagatorConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/OauthErrorResponseException.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/InterceptUrlMapRule.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationFailureReason.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationProvider.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/rules/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/validator/RefreshTokenValidator.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthorizationException.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/CookiePriorToLoginPersistence.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/ForbiddenRedirectConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/RedirectConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/errors/OauthErrorResponseExceptionHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/MapClaims.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/propagation/TokenPropagationHttpClientFilter.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/event/AccessTokenGeneratedEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/event/TokenValidatedEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/token/DefaultRolesFinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/config/UnauthorizedRedirectConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/AuthenticationException.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-
jar/io/micronaut/security/authentication/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-

jar/io/micronaut/security/errors/ErrorResponse.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/config/AuthenticationModeConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/x509/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/config/RedirectConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/AuthenticationExceptionHandler.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/AuthenticationArgumentBinder.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/CookieBasedAuthenticationModeCondition.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/token/package-info.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/token/propagation/TokenPropagationConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/endpoints/LoginControllerConfigurationProperties.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/config/InterceptUrlMapPattern.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/authentication/ClientAuthentication.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/endpoints/LoginControllerConfiguration.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/event/LoginSuccessfulEvent.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/config/InterceptUrlMapConverter.java
* /opt/cola/permits/1331474204_1653068520.815044/0/micronaut-security-3-5-0-sources-jar/io/micronaut/security/token/propagation/TokenPropagationConfiguration.java

1.135 fabric8-::-kubernetes-model-::- networking 4.10.3

1.135.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.136 micronaut-tracing 4.1.1

1.136.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2022 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/interceptor/TraceInterceptor.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/HttpHeadersTextMap.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/TracingPublisher.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/TracingPublisherUtils.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/annotation/SpanTag.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/TracingExclusionsConfiguration.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/OpenTracingClientFilter.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/OpenTracingInvocationInstrumenter.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/TracingObserver.java
```

```
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/kotlin/HttpCoroutineTracingDispatcherFactory.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/annotation/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/MdcInstrumenter.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/annotation/NewSpan.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/interceptor/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/TracingCorePublisher.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/TraceRequestAttributes.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/AbstractOpenTracingFilter.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/kotlin/CoroutineTracingDispatcher.kt
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/ScopePropagationPublisher.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/TracingInvocationInstrumenterFactory.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/package-info.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/util/ThreadTracingInvocationInstrumenterFactory.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/annotation/ContinueSpan.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/DefaultTracer.java
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-
jar/io/micronaut/tracing/instrument/http/OpenTracingServerFilter.java
```

No license file was found, but licenses were detected in source scan.

```
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1331473843_1653512495.747282/0/micronaut-tracing-core-4-1-1-sources-jar/META-
INF/native-image/io.micronaut.tracing/tracing-core/native-image.properties
```

1.137 wire-protocol-buffer-runtime 3.2.2

1.137.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2015 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
 * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/ProtoAdapter.kt
 * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/internal/MutableOnWriteList.kt
 * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/MessageSerializedForm.kt
 * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/ProtoAdapter.kt
 * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/internal/FieldBinding.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2013 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/RuntimeEnumAdapter.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/internal/RuntimeMessageAdapter.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/Message.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/internal/InternalJvm.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/internal/ImmutableList.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/FieldEncoding.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/Wire.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/WireEnum.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/Message.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/WireField.kt

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2019 Square Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/Service.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/internal/Internal.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/MessageSource.kt
- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/WireRpc.kt

```
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/MessageSink.kt
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/internal/-Platform.kt
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/MessageSink.kt
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/MessageSource.kt
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/jvmMain/com/squareup/wire/internal/-Platform.kt
No license file was found, but licenses were detected in source scan.
```

```
/*
 * Copyright 2013 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
// Copyright 2013 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/ProtoWriter.kt
No license file was found, but licenses were detected in source scan.
```

```
/*
 * Copyright 2020 Square Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
```

* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/AnyMessage.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2019 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-
jar/commonMain/com/squareup/wire/internal/Util.kt
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 Square Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/EnumAdapter.kt
 - * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/EnumAdapter.kt
 - * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/jvmMain/com/squareup/wire/AndroidMessage.kt
- No license file was found, but licenses were detected in source scan.

```
// Copyright 2013 Google Inc. All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// * Neither the name of Google Inc. nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

- * /opt/cola/permits/1337069434_1654201885.1354642/0/wire-runtime-3-2-2-sources-jar/commonMain/com/squareup/wire/ProtoReader.kt

1.138 dropwizard 2.0.18

1.138.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but

excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its

distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise,

unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.139 mockito 4.3.1

1.139.1 Available under license :

The MIT License

Copyright (c) 2007 Mockito contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.140 alpn-api 1.1.3.v20160715

1.140.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved. This program and the accompanying materials
// are made available under the terms of the Eclipse Public License v1.0
// and Apache License v2.0 which accompanies this distribution.
// The Eclipse Public License is available at
// The Apache License v2.0 is available at
// You may elect to redistribute this code under either of these licenses.
```

Found in path(s):

```
* /opt/cola/permits/1162865313_1620924138.73/0/alpn-api-1-1-3-v20160715-sources-1-
jar/org/eclipse/jetty/alpn/ALPN.java
```

1.141 slf4j-api-module 1.7.29

1.141.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/**
 * Copyright (c) 2004-2011 QOS.ch
 * All rights reserved.
 *
 * Permission is hereby granted, free of charge, to any person obtaining
 * a copy of this software and associated documentation files (the
 * "Software"), to deal in the Software without restriction, including
 * without limitation the rights to use, copy, modify, merge, publish,
 * distribute, sublicense, and/or sell copies of the Software, and to
 * permit persons to whom the Software is furnished to do so, subject to
 * the following conditions:
 *
 * The above copyright notice and this permission notice shall be
 * included in all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
 * EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
 * MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
 * NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE
 * LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION
 * OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/BasicMarker.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/MessageFormatter.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/BasicMDCAdapter.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/FormattingTuple.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/IMarkerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/spi/LocationAwareLogger.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/MDC.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/Util.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/helpers/SubstituteLogger.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-
jar/org/slf4j/LoggerFactory.java
```

* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/impl/StaticMDCBinder.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/spi/LoggerFactoryBinder.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/Logger.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/impl/StaticMarkerBinder.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/SubstituteLoggerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/NOPLogger.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/BasicMarkerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/NOPMDCAadapter.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/spi/MarkerFactoryBinder.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/ILoggerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/NOPLoggerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/NamedLoggerBase.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/spi/MDCAdapter.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/MarkerFactory.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/impl/StaticLoggerBinder.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/helpers/MarkerIgnoringBase.java
* /opt/ws_local/PERMITS_SQL/1014221182_1593546679.56/0/slf4j-api-1-7-29-sources-jar/org/slf4j/Marker.java

1.142 jcl-1-2-implemented-over-slf4j 1.7.30

1.142.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License, Version 2.0</name>

Found in path(s):

* /opt/cola/permits/1049595338_1647856010.49/0/jcl-over-slf4j-1-7-30-1-jar/META-INF/maven/org.slf4j/jcl-over-slf4j/pom.xml

1.143 dropwizard-jersey-support 2.0.18

1.143.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.144 apache-httpmime 4.5.5

1.144.1 Available under license :

Apache HttpClient Mime
Copyright 1999-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.145 config-com-typesafe-config 1.3.1

1.145.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.146 micronaut-email 1.2.0

1.146.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2021 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/StringBody.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/EmailDecorator.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/DefaultAsyncEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/FromDecorator.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/validation/RecipientsUtils.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/Email.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/EmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/Attachment.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-
jar/io/micronaut/email/validation/AnyRecipient.java
```

* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/DefaultEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/configuration/FromConfigurationProperties.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/MultipartBody.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/validation/AnyRecipientConstraintValidatorFactory.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/EmailException.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/BodyType.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/Body.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/TransactionalEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/TrackLinks.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/Contact.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/AsyncTransactionalEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/validation/Recipients.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/configuration/FromConfiguration.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/EmailComposer.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/AsyncEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/AbstractTransactionalEmailSender.java
* /opt/cola/permits/1340031375_1654618487.6340535/0/micronaut-email-1-2-0-sources-jar/io/micronaut/email/validation/EmailMessages.java

1.147 jakarta-expression-language-3.0-implementation 4.0.2

1.147.1 Available under license :

Notices for Jakarta Expression Language

This content is produced and maintained by the Jakarta Expression Language project.

* Project home: <https://projects.eclipse.org/projects/ee4j.el>

Trademarks

Jakarta Expression Language is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/el-ri>

Third-party Content

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any

exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

- a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and
- b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 - i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 - iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published,

Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free

Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either

verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR

OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this

exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.148 dropwizard-servlet-support 2.0.18

1.148.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.149 fabric8-::-kubernetes-model-::-policy

4.10.3

1.149.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.150 apache-xerces2-j 2.12.2

1.150.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====

Apache Xerces Java
Copyright 1999-2022 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the
Apache Software Foundation that were originally developed at iClick, Inc.,
software copyright (c) 1999.

1.151 micronaut-data 3.3.0

1.151.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2022 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-  
jar/io/micronaut/data/operations/HintsCapableRepository.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/naming/NamingStrategies.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/AutoPopulated.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/repeatable/WhereSpecifications.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PreUpdate.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/SaveAllReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/QueryParameter.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/BatchOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/GenericRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/NamingStrategy.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/CountReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/AssociationUtils.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PostRemove.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
```

jar/io/micronaut/data/intercept/annotation/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Where.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindSliceInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/repeatable/TypeDefinitions.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/sql/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Pageable.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/PersistentProperty.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/QueryBuilder.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PreRemove.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/DeleteOneAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/reactive/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/CrudRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/UpdateEntityReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/EntityEventMapping.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Slice.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/UpdateInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Query.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/AbstractPersistentEntity.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/SaveEntityAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Association.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/DataTransformer.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/repeatable/JoinSpecifications.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/reactive/RxJavaCrudRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/intercept/async/FindSliceAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/SaveOneInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/PersistentEntity.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/jpa/JpaSpecificationExecutor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/AssociationQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Relation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/FindAllReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/DataType.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/PrimaryRepositoryOperations.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/QueryHint.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Id.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/UpdateAllEntitiesInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/DefaultStoredDataOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/RuntimeEmbedded.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/StoredDataOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/SaveOneAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/NonUniqueResultException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/PreparedQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/sql/SqlQueryConfiguration.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/async/AsyncRepositoryOperations.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/reactive/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/RepositoryConfiguration.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Repository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/MappedEntity.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/intercept/async/FindOneAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/jpa/criteria/SpecificationComposition.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/DefaultQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Version.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/DataAccessResourceFailureException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/SaveEntityReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/JoinPath.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/ExistsByInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/FindPageReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/sql/Dialect.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/DefaultPageable.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/RepositoryOperations.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/async/AsyncCrudRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindOptionalInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/Criteria.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/QueryResult.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/FindOneReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/IterableResultInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/StoredQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Embedded.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Page.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindStreamInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/model/runtime/convert/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/PageableRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/async/AsyncCapableRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/DeleteBatchOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/QueryModel.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/ExistsByAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/DeleteAllAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/factory/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/EmptyResultException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/UpdateOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/reactive/ReactiveStreamsCrudRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/OptimisticLockException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/MappedProperty.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Expandable.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/SaveEntityInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PrePersist.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/ExistsByReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/RuntimePersistentProperty.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/UpdateAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/DataIntroductionAdvice.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/UpdateAllEntitiesReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/annotation/Embeddable.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/EntityInstanceOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/jpa/kotlin/CoroutineJpaSpecificationExecutor.kt
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/FindByIdAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/SaveAllAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/DataAccessException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/factory/Projections.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/EmbeddedId.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Join.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/UpdateEntityAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/repeatable/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/Sort.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/DateCreated.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PostLoad.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/AbstractSqlLikeQueryBuilder.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/ProjectionList.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/DateUpdated.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/annotation/DataMethod.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/CountAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/DefaultSort.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/DeleteOneInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/exceptions/MappingException.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/AbstractPreparedDataOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/annotation/TypeDef.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/UpdateReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/RuntimeAssociation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/UpdateBatchOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/Transient.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/reactive/ReactiveCapableRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindAllInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/FindPageAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/DeleteOneReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/InsertBatchOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/RuntimePersistentEntity.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindPageInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/GeneratedValue.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/sql/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/PersistentElement.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/DataInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/UpdateEntityInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/naming/NamingStrategy.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/UpdateAllEntriesAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/FindByIdReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/DeleteAllInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/reactive/ReactiveRepositoryOperations.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/DefaultSlice.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PostPersist.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/model/runtime/PreparedDataOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/FindByIdInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/TypeRole.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/async/FindAllAsyncInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/sql/SqlQueryBuilder.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/DefaultPage.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/RepositoryMethodKey.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/SaveAllInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/DeleteAllReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/jpa/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/InsertOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/builder/jpa/JpaQueryBuilder.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/DeleteOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/PagedQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/repository/async/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/event/PostUpdate.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/annotation/sql/SqlMembers.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/factory/Restrictions.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/naming/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/CountInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/query/DefaultProjectionList.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/operations/async/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/model/runtime/EntityOperation.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-
jar/io/micronaut/data/intercept/reactive/SaveOneReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-

jar/io/micronaut/data/annotation/package-info.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/intercept/FindOneInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/intercept/reactive/FindSliceReactiveInterceptor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/annotation/repeatable/QueryHints.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/intercept/annotation/DataMethodQueryParameter.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/intercept/reactive/package-info.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2021 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/IExpression.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/EntityEventContext.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/QueryResultPersistentEntityCriteriaQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/operations/reactive/BlockingReactorRepositoryOperations.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/DisjunctionPredicate.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/kotlin/CoroutineCrudRepository.kt
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractPersistentEntityCriteriaQuery.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/SelectionVisitor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/AggregateExpression.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityCriteriaQuery.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/ParameterExpressionImpl.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractCriteriaBuilder.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/util/Joiner.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/CompoundSelection.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/runtime/convert/AttributeConverter.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityCriteriaBuilder.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/CriteriaUtils.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityPath.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/IPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PersistentPropertyInValuesPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityFrom.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityCriteriaUpdate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/runtime/AttributeConverterRegistry.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/QueryEventContext.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractPersistentEntityCriteriaUpdate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/PersistentPropertyPath.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PostUpdateEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PersistentPropertyUnaryPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/operations/reactive/ReactorReactiveRepositoryOperations.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityJoin.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/criteria/DeleteSpecification.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PostPersistEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/criteria/PredicateSpecification.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/reactive/ReactorJpaSpecificationExecutor.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/IdExpression.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/annotation/Indexes.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PersistentPropertyInPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/PersistentAssociationPath.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/annotation/Index.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/AbstractPersistentPropertyExpression.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/runtime/PropertyAutoPopulator.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/PersistentPropertyOrder.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PersistentPropertyBetweenPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PreRemoveEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PersistentPropertyBinaryPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/runtime/RuntimeEntityRegistry.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityRoot.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/query/BindingContextImpl.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/async/AsyncJpaSpecificationExecutor.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/query/BindingParameter.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PredicateBinaryOp.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractPersistentPropertyPath.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/ISelection.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/PredicateVisitable.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/criteria/QuerySpecification.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/query/QueryModelSelectionVisitor.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/LiteralExpression.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/criteria/UpdateSpecification.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractPersistentEntityCriteriaDelete.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/AbstractPersistentEntityJoinSupport.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/ConjunctionPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/EntityEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/operations/reactive/ReactorReactiveCapableRepository.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/PersistenceEventException.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/runtime/QueryParameterBinding.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PostRemoveEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/ExpressionBinaryPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/AliasedSelection.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/AbstractNumericalPersistentPropertyExpression.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/QueryModelPersistentEntityCriteriaQuery.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/PredicateUnaryOp.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/query/builder/QueryParameterBinding.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PrePersistEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/SelectionVisitable.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/jpa/reactive/ReactiveStreamsJpaSpecificationExecutor.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/selection/AggregateType.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/AbstractPersistentPropertyPredicate.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/PredicateVisitor.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/PersistenceEventContext.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentEntityCriteriaDelete.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/event/listeners/PreUpdateEventListener.java

* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/repository/reactive/ReactorCrudRepository.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/query/QueryModelPredicateVisitor.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/NegatedPredicate.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/impl/predicate/AbstractPredicate.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentAssociationPath.java
* /opt/cola/permits/1331474323_1653513905.8667567/0/micronaut-data-model-3-3-0-sources-jar/io/micronaut/data/model/jpa/criteria/PersistentPropertyPath.java

1.152 pact-jvm-consumer 4.0.10

1.152.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.153 micronaut-serialization 1.0.1

1.153.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2022 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/util/CustomizableSerializer.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/util/CustomizableDeserializer.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2021 original authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/util/NullableSerde.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/ObjectMapper.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/annotation/SerdeImport.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/config/naming/LowerDotCaseStrategy.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/Serde.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/DeserializerLocator.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/config/naming/SnakeCaseStrategy.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/NamingStrategyLocator.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/Deserializer.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/config/naming/KebabCaseStrategy.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-jar/io/micronaut/serde/config/naming/PropertyNamingStrategy.java

* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-

jar/io/micronaut/serde/SerializerLocator.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/exceptions/SerdeException.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/Encoder.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/naming/UpperCamelCaseStrategyWithSpaces.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/annotation/SerdeConfig.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/exceptions/InvalidPropertyFormatException.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/reference/AbstractPropertyReferenceManager.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/SerdeIntrospections.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/DeserializationConfiguration.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/SerdeRegistry.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/Decoder.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/SerdeConfiguration.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/util/NullableDeserializer.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/naming/LowerCaseStrategy.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/exceptions/InvalidFormatException.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/naming/LowerCamelCaseStrategy.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/UpdatingDeserializer.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/naming/UpperCamelCaseStrategy.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/reference/PropertyReference.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/naming/IdentityStrategy.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/Serializer.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/reference/SerializationReference.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/config/SerializationConfiguration.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-
jar/io/micronaut/serde/annotation/Serdeable.java
* /opt/cola/permits/1331474218_1653513142.445036/0/micronaut-serde-api-1-0-1-sources-

1.154 apache-groovy 3.0.9

1.154.1 Available under license :

ASM License

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2011 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Groovy

Copyright 2003-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

This product includes/uses ANTLR2 (<http://www.antlr2.org/>)
developed by Terence Parr 1989-2006

This product includes/uses ANTLR4 (<https://github.com/antlr/antlr4>)

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

ANTLR 2 License

Antlr2 is released in the public domain.
See licenses/antlr2-license.txt for details.

ANTLR 4 License

Antlr4 is released under a BSD 3-clause license.
See licenses/antlr4-license.txt for details.

ASM License

ASM uses a 3-clause BSD license. For details, see licenses/asm-license.txt.
ANTLR 2 License

We reserve no legal rights to the ANTLR--it is fully in the public domain. An individual or company may do whatever they wish with source code distributed with ANTLR or the code generated by ANTLR, including the incorporation of ANTLR, or its output, into commercial software.

We encourage users to develop software with ANTLR. However, we do ask that credit is given to us for developing ANTLR. By "credit", we mean that if you use ANTLR or incorporate any source code into one of your programs (commercial product, research project, or otherwise) that you acknowledge this fact somewhere in the documentation, research report, etc... If you like ANTLR and have developed a nice tool with the output, please mention that you developed it using ANTLR. In addition, we ask that the headers remain intact in our source code. As long as these guidelines are kept, we expect to continue enhancing this system and expect to make other tools available as they are completed.

In countries where the Public Domain status of the work may not be valid, the author grants a copyright licence to the general public to deal in the work without restriction and permission to sublicense derivatives under the terms of any (OSI approved) Open Source licence.

The Python parser generator code under antlr/actions/python/ is covered by the 3-clause BSD licence (this part is included in the binary JAR files); the run-time part under lib/python/ is covered by the GNU GPL, version 3 or later (this part is not included in the binary JAR files). See [1] for the full details.

<https://bugs.debian.org/cgi-bin/bugreport.cgi?bug=750643#80%22>

ANTLR 4 License

[The "BSD 3-clause license"]

Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.155 ktor-http-cio 1.2.6

1.155.1 Available under license :

/*

* Copyright 2014-2019 JetBrains s.r.o and contributors. Use of this source code is governed by the Apache 2.0 license.

*/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.156 dropwizard-validation-support 2.0.18

1.156.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.157 ktor-utils-jvm 1.2.6

1.157.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.158 opentelemetry-java---

io.opentelemetry:opentelemetry-sdk-metrics

1.6.0-alpha

1.158.1 Available under license :

c-libutl 20160225

c-libutl License

...

This software is distributed under the terms of the BSD license.

== BSD LICENSE =====

(C) 2009 by Remo Dentato (rdentato@gmail.com)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<http://opensource.org/licenses/bsd-license.php>

...

IAIK (Institute for Applied Information Processing and Communication) PKCS#11 wrapper files v1

IAIK License

<pre>

Copyright (c) 2002 Graz University of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by IAIK of Graz University of Technology."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Graz University of Technology" and "IAIK of Graz University of Technology" must not be used to endorse or promote products derived from this software without prior written permission.

5. Products derived from this software may not be called "IAIK PKCS Wrapper", nor may "IAIK" appear in their name, without prior written permission of Graz University of Technology.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE LICENSOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

This library is free software; you can redistribute it and/or modify it under the terms of the Modified BSD License.

The complete text of the license is available at the Documentation/licenses/COPYING.BSD-3 file.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser
General Public License, and the "GNU GPL" refers to version 3 of the GNU
General Public License.

"The Library" refers to a covered work governed by this License,
other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided
by the Library, but which is not otherwise based on the Library.
Defining a subclass of a class defined by the Library is deemed a mode
of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an
Application with the Library. The particular version of the Library
with which the Combined Work was made is also called the "Linked
Version".

The "Minimal Corresponding Source" for a Combined Work means the
Corresponding Source for the Combined Work, excluding any source code
for portions of the Combined Work that, considered in isolation, are
based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the
object code and/or source code for the Application, including any data
and utility programs needed for reproducing the Combined Work from the
Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License
without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or
- b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the

copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

SPECIAL EXCEPTION GRANTED BY COPYRIGHT HOLDERS

As a special exception, copyright holders give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you must extend this exception to your version of the library.

Double-conversion v1.1.5

Double-conversion License

<https://raw.githubusercontent.com/google/double-conversion/master/LICENSE>

<pre>

Copyright 2006-2011, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright

notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

Apache Xerces v2.12.1

Apache Xerces Notice

<pre>

```
=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====
```

Apache Xerces Java
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

</pre>

Apache 2.0 License

<pre>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including

the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each

Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

</pre>

Eastman Kodak Company: Portions of color management and imaging software

Eastman Kodak Notice

<pre>

Portions Copyright Eastman Kodak Company 1991-2003

</pre>

End User License Agreement Missing

Apache HttpClient Cache

Copyright 2010-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====

Apache Xerces Java

Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

- a) under this License, provided that you make a good faith effort to

ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to

recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the

Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

SPECIAL EXCEPTION GRANTED BY IMATIX

As a special exception, iMatix gives you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you must extend this exception to your version of the library.

Parts of the software are licensed under the MIT (X11) license as follows:

Copyright (c) 2007-2010 Contributors as listed in AUTHORS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES

OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache Xalan v2.7.2

Apache Xalan Notice

<pre>

```
=====
====
== NOTICE file corresponding to the section 4d of the Apache License, Version 2.0, ==
== in this case for the Apache Xalan distribution. ==
=====
====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Specifically, we only include the XSLTC portion of the source from the Xalan distribution.
The Xalan project has two processors: an interpretive one (Xalan Interpretive) and a compiled one (The XSLT Compiler (XSLTC)). We *only* use the XSLTC part of Xalan; We use the source from the packages that are part of the XSLTC sources.

Portions of this software was originally based on the following:

- software copyright (c) 1999-2002, Lotus Development Corporation., <http://www.lotus.com>.
- software copyright (c) 2001-2002, Sun Microsystems., <http://www.sun.com>.
- software copyright (c) 2003, IBM Corporation., <http://www.ibm.com>.
- voluntary contributions made by Ovidiu Predescu (ovidiu@cup.hp.com) on behalf of the Apache Software Foundation and was originally developed at Hewlett Packard Company.

</pre>

Apache 2.0 License

<pre>

```
        Apache License
        Version 2.0, January 2004
        http://www.apache.org/licenses/
```

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

JLEX COPYRIGHT NOTICE, LICENSE AND DISCLAIMER.

Copyright 1996-2003 by Elliot Joel Berk and C. Scott Ananian
Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software. The portions of JLex output which are hard-coded into the JLex source code are (naturally) covered by this same license.

</pre>

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts.

OpenSSL License

/* =====

* Copyright (c) 1998-2019 The OpenSSL Project. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

*

* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.

*
 * 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).

- * The implementation was written so as to conform with Netscapes SSL.
- *
- * This library is free for commercial and non-commercial use as long as
- * the following conditions are adhered to. The following conditions
- * apply to all code found in this distribution, be it the RC4, RSA,
- * lhash, DES, etc., code; not just the SSL code. The SSL documentation
- * included with this distribution is covered by the same copyright terms
- * except that the holder is Tim Hudson (tjh@cryptsoft.com).
- *
- * Copyright remains Eric Young's, and as such any Copyright notices in
- * the code are not to be removed.
- * If this package is used in a product, Eric Young should be given attribution
- * as the author of the parts of the library used.
- * This can be in the form of a textual message at program startup or
- * in documentation (online or textual) provided with the package.
- *
- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- * 1. Redistributions of source code must retain the copyright
- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in the
- * documentation and/or other materials provided with the distribution.
- * 3. All advertising materials mentioning features or use of this software
- * must display the following acknowledgement:
- * "This product includes cryptographic software written by
- * Eric Young (eay@cryptsoft.com)"
- * The word 'cryptographic' can be left out if the routines from the library
- * being used are not cryptographic related :-).
- * 4. If you include any Windows specific code (or a derivative thereof) from
- * the apps directory (application code) you must include an acknowledgement:
- * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
- *
- * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
- * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
- * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
- * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
- * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
- * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
- * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
- * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
- * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
- * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
- * SUCH DAMAGE.
- *
- * The licence and distribution terms for any publically available version or
- * derivative of this code cannot be changed. i.e. this code cannot simply be

* copied and put under another distribution licence

* [including the GNU Public Licence.]

*/

xml-commons/java/external/README.dom.txt \$Id: README.dom.txt 477038 2006-11-20 04:40:36Z mrglavas \$

HEAR YE, HEAR YE!

All of the .java software and associated documentation about the DOM in this repository are distributed under the license from the W3C, which is provided herein.

LICENSE.dom-software.txt covers all software from the W3C including the following items in the xml-commons project:

xml-commons/java/external/src/org/w3c
and all subdirectories

(Note: SAC (The Simple API for CSS) has been published under an older version of the W3C license. The original license file is LICENSE.sac.html.)

LICENSE.dom-documentation.txt covers all documentation from the W3C including the following items in the xml-commons project:

xml-commons/java/external/xdocs/dom
and all subdirectories

The actual DOM Java Language Binding classes in xml-commons came from:
<http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.html>

The specification of DOM Level 3's various parts is at:
<http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/>
<http://www.w3.org/TR/2004/REC-DOM-Level-3-LS-20040407/>
<http://www.w3.org/TR/2004/NOTE-DOM-Level-3-XPath-20040226/>

The specification of DOM Level 2's various parts is at:
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Traversal-Range-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Views-20001113/>

The specification of DOM Level 1's various parts is at:
<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/level-one-html.html>

Links to all available W3C DOM Java Bindings can be found at:
<http://www.w3.org/DOM/DOMTR>

The actual classes of The Simple API for CSS (SAC) came from:

<http://www.w3.org/Style/CSS/SAC/>

<http://www.w3.org/2002/06/sacjava-1.3.zip>

The actual DOM Java Language Binding classes for SMIL came from:

<http://dev.w3.org/cvsweb/java/classes/org/w3c/dom/smil/>

(both ElementTimeControl.java and TimeEvent.java were taken at revision 1.1)

The actual DOM Java Language Binding classes for SVG 1.1 came from:

<http://www.w3.org/TR/SVG11/java.html>

PC/SC Lite v1.8.26

PC/SC Lite License

<pre>

Copyright (c) 1999-2003 David Corcoran <corcoran@linuxnet.com>

Copyright (c) 2001-2011 Ludovic Rousseau <ludovic.rousseau@free.fr>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Changes to this license can be made only by the copyright author with explicit written consent.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

Apache Commons IO
Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

OPENJDK ASSEMBLY EXCEPTION

The OpenJDK source code made available by Oracle America, Inc. (Oracle) at openjdk.java.net ("OpenJDK Code") is distributed under the terms of the GNU General Public License <<http://www.gnu.org/copyleft/gpl.html>> version 2 only ("GPL2"), with the following clarification and special exception.

Linking this OpenJDK Code statically or dynamically with other code is making a combined work based on this library. Thus, the terms and conditions of GPL2 cover the whole combination.

As a special exception, Oracle gives you permission to link this OpenJDK Code with certain code licensed by Oracle as indicated at <http://openjdk.java.net/legal/exception-modules-2007-05-08.html> ("Designated Exception Modules") to produce an executable, regardless of the license terms of the Designated Exception Modules, and to copy and distribute the resulting executable under GPL2, provided that the Designated Exception Modules continue to be governed by the licenses under which they were offered by Oracle.

As such, it allows licensees and sublicensees of Oracle's GPL2 OpenJDK Code to build an executable that includes those portions of necessary code that Oracle could not provide under GPL2 (or that Oracle has provided under GPL2 with the Classpath exception). If you modify or add to the OpenJDK code, that new GPL2 code may still be combined with Designated Exception Modules if the new code is made subject to this exception by its copyright holder.

Independent JPEG Group: JPEG release 6b

JPEG License

<pre>

Must reproduce following license in documentation and/or other materials provided with distribution:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

(1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA. ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual distribution terms of the Free Software Foundation; principally, that you must include source code if you redistribute it.

(See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable.

The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding

cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.) So far as we are aware, there are no patent restrictions on the remaining code.

The IJG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

</pre>

```
=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache xml-commons xml-apis ==
== distribution. ==
=====
```

Apache XML Commons XML APIs
Copyright 1999-2009 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- software copyright (c) 2000 World Wide Web Consortium, <http://www.w3.org>
DOM Level 3 Core Specification v1.0

W3C License
<pre>

W3C SOFTWARE NOTICE AND LICENSE

<http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms

and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

- 1.The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- 2.Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- 3.Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS,COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION. The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

</pre>

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty

percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly

granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer

and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the

original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do

not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the

sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

The Netty Project

=====

Please visit the Netty web site for more information:

* <http://netty.io/>

Copyright 2016 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a forked and modified version of Tomcat Native

* LICENSE:

* [license/LICENSE.tomcat-native.txt](#) (Apache License 2.0)

* HOMEPAGE:

* <http://tomcat.apache.org/native-doc/>

* <https://svn.apache.org/repos/asf/tomcat/native/>

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:

* [license/LICENSE.mvn-wrapper.txt](#) (Apache License 2.0)

* HOMEPAGE:

* <https://github.com/takari/maven-wrapper>

This product contains small piece of code to support AIX, taken from netbsd.

* LICENSE:

* [license/LICENSE.aix-netbsd.txt](#) (OpenSSL License)

* HOMEPAGE:

* <https://ftp.netbsd.org/pub/NetBSD/NetBSD-current/src/crypto/external/bsd/openssl/dist>

This product contains code from boringssl.

* LICENSE (Combination ISC and OpenSSL license)

* [license/LICENSE.boringssl.txt](#) (Combination ISC and OpenSSL license)

* HOMEPAGE:

* <https://boringsssl.googleusercontent.com/boringsssl/>

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced

by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or

in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to

distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent

obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of

all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or

FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author Gnomovision comes
with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free
software, and you are welcome to redistribute it under certain conditions;
type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
signature of Ty Coon, 1 April 1989
```

```
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

```
## Dynalink v.5
```

```
### Dynalink License
```

```
<pre>
```

Copyright (c) 2009-2013, Attila Szegedi

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
</pre>
```

```
xml-commons/java/external/LICENSE.dom-software.txt $Id: LICENSE.dom-software.txt 734314 2009-01-14 03:33:27Z mrglavas $
```

This license came from: <http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.zip>
(COPYRIGHT.html)

W3C SOFTWARE NOTICE AND LICENSE

Copyright 2004 World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University).
All Rights Reserved.

The DOM bindings are published under the W3C Software Copyright Notice and License. The software license requires "Notice of any changes or modifications to the W3C files, including the date changes were made." Consequently, modified versions of the DOM bindings must document that they do not conform to the W3C standard; in the case of the IDL definitions, the pragma prefix can no longer be 'w3c.org'; in the case of the Java language binding, the package names can no longer be in the 'org.w3c' package.

Note: The original version of the W3C Software Copyright Notice and License could be found at <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR

CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission.

Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium:

<https://cla.developers.google.com/cla>

Files in third_party/ have their own licenses, as described therein. The MIT license, for third_party/flat, which, unlike other third_party directories, is compiled into non-test libraries, is included below.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

The following are Google-internal bug numbers where explicit permission from some authors is recorded for use of their work. (This is purely for our own record keeping.)

27287199

27287880

27287883

OpenSSL License

/* =====

* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the

* distribution.

*

* 3. All advertising materials mentioning features or use of this

* software must display the following acknowledgment:

* "This product includes software developed by the OpenSSL Project

* for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"

*

* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to

* endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *

* 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *

* 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *

* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *

* This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

 /* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *

* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
* must display the following acknowledgement:
* "This product includes cryptographic software written by
* Eric Young (eay@cryptsoft.com)"
* The word 'cryptographic' can be left out if the rouines from the library
* being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
* the apps directory (application code) you must include an acknowledgement:
* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
ISC license used for completely new code in BoringSSL:
/* Copyright (c) 2015, Google Inc.
*
* Permission to use, copy, modify, and/or distribute this software for any
* purpose with or without fee is hereby granted, provided that the above
* copyright notice and this permission notice appear in all copies.

*

* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
* SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
* OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
* CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

The code in third_party/flat carries the MIT license:

Copyright (c) 2015-2016 the fiat-crypto authors (see

<https://github.com/mit-plv/flat-crypto/blob/master/AUTHORS>).

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Licenses for support code

Parts of the TLS test suite are under the Go license. This code is not included in BoringSSL (i.e. libcrypto and libssl) when compiled, however, so distributing code linked against BoringSSL does not trigger this license:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BoringSSL uses the Chromium test infrastructure to run a continuous build, trybots etc. The scripts which manage this, and the script for generating build metadata, are under the Chromium license. Distributing code linked against BoringSSL does not trigger this license.

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Commons Math

Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

=====
The LinearConstraint, LinearObjectiveFunction, LinearOptimizer,
Relationship, SimplexSolver and SimplexTableau classes in package
org.apache.commons.math.optimization.linear include software developed by
Benjamin McCann (<http://www.benmccann.com>) and distributed with
the following copyright: Copyright 2009 Google Inc.
=====

This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

The LevenbergMarquardtOptimizer class in package org.apache.commons.math.optimization.general includes software translated from the lmdcr, lmdcr and qrsolv Fortran routines from the Minpack package
Minpack Copyright Notice (1999) University of Chicago. All rights reserved

The GraggBulirschStoerIntegrator class in package org.apache.commons.math.ode.nonstiff includes software translated from the odex Fortran routine developed by E. Hairer and G. Wanner.
Original source copyright:
Copyright (c) 2004, Ernst Hairer

The EigenDecompositionImpl class in package org.apache.commons.math.linear includes software translated from some LAPACK Fortran routines. Original source copyright:
Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

The MersenneTwister class in package org.apache.commons.math.random includes software translated from the 2002-01-26 version of the Mersenne-Twister generator written in C by Makoto Matsumoto and Takuji Nishimura. Original source copyright:
Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,
All rights reserved

The complete text of licenses and disclaimers associated with the the original sources enumerated above at the time of code translation are in the LICENSE.txt file.

Mozilla Public Suffix List

Public Suffix Notice

^^^

You are receiving a copy of the Mozilla Public Suffix List in the following file: <java-home>/lib/security/public_suffix_list.dat. The terms of the Oracle license do NOT apply to this file; it is licensed under the Mozilla Public License 2.0, separately from the Oracle programs you receive. If you do not wish to use the Public Suffix List, you may remove the <java-home>/lib/security/public_suffix_list.dat file.

The Source Code of this file is available under the Mozilla Public License, v. 2.0 and is located at

https://raw.githubusercontent.com/publicsuffix/list/cbbba1d234670453df9c930dfbf510c0474d4301/public_suffix_list.dat.

If a copy of the MPL was not distributed with this file, you can obtain one at <https://mozilla.org/MPL/2.0/>.

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

...

MPL v2.0

...

Mozilla Public License Version 2.0

=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software; or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

(a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and

(b) You may distribute such Executable Form under the terms of this

License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute

or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
*
*
* 6. Disclaimer of Warranty
* -----
*
* Covered Software is provided under this License on an "as is"
* basis, without warranty of any kind, either expressed, implied, or
* statutory, including, without limitation, warranties that the
* Covered Software is free of defects, merchantable, fit for a
* particular purpose or non-infringing. The entire risk as to the
* quality and performance of the Covered Software is with You.
* Should any Covered Software prove defective in any respect, You
* (not any Contributor) assume the cost of any necessary servicing,
* repair, or correction. This disclaimer of warranty constitutes an
* essential part of this License. No use of any Covered Software is
* authorized under this License except under this disclaimer.
```

*

*

* 7. Limitation of Liability *
 * ----- *

*

* Under no circumstances and under no legal theory, whether tort *
 * (including negligence), contract, or otherwise, shall any *
 * Contributor, or anyone who distributes Covered Software as *
 * permitted above, be liable to You for any direct, indirect, *
 * special, incidental, or consequential damages of any character *
 * including, without limitation, damages for lost profits, loss of *
 * goodwill, work stoppage, computer failure or malfunction, or any *
 * and all other commercial damages or losses, even if such party *
 * shall have been informed of the possibility of such damages. This *
 * limitation of liability shall not apply to liability for death or *
 * personal injury resulting from such party's negligence to the *
 * extent applicable law prohibits such limitation. Some *
 * jurisdictions do not allow the exclusion or limitation of *
 * incidental or consequential damages, so this exclusion and *
 * limitation may not apply to You. *

*

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <https://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as

defined by the Mozilla Public License, v. 2.0.

...

Apache Commons Byte Code Engineering Library (BCEL) Version 6.4.1

Apache Commons BCEL Notice

<pre>

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                    ==
== in this case for the Apache Commons BCEL distribution. ==
=====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

</pre>

Apache 2.0 License

<pre>

```
        Apache License
        Version 2.0, January 2004
        http://www.apache.org/licenses/
```

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,

including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf

of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

</pre>

```
## Mesa 3-D Graphics Library v21.0.3
```

```
### Mesa License
```

```
...
```

```
Copyright (C) 1999-2007 Brian Paul All Rights Reserved.
```

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included

in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Attention, Contributors

When contributing to the Mesa project you must agree to the licensing terms of the component to which you're contributing.

The following section lists the primary components of the Mesa distribution and their respective licenses.

Mesa Component Licenses

Component	Location	License
Main Mesa code	src/ mesa/	MIT
Device drivers	src/ mesa/ drivers/ *	MIT, generally
Gallium code	src/ gallium/	MIT
Ext headers	GL/ glxext.h	Khronos
	GL/ glxext.h	Khronos
	GL/ wglxext.h	Khronos
	KHR/ khrplatform.h	Khronos

include/GL/gl.h :

Mesa 3-D graphics library

Copyright (C) 1999-2006 Brian Paul All Rights Reserved.

Copyright (C) 2009 VMware, Inc. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the

Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
include/GL/glex.h
include/GL/glxext.h
include/GL/wglxext.h :
```

Copyright (c) 2013 - 2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

```
include/KHR/khrplatform.h :
```

Copyright (c) 2008 - 2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

...

The FreeType Project: Freetype v2.10.4

FreeType Notice

...

FreeType comes with two licenses from which you can choose the one which fits your needs best.

The FreeType License (FTL) is the most commonly used one. It is a BSD-style license with a credit clause and thus compatible with the GNU Public License (GPL) version 3, but not with the GPL version 2.

The GNU General Public License (GPL), version 2. Use it for all projects which use the GPLv2 also, or which need a license compatible to the GPLv2.

...

FreeType License

...

The FreeType Project LICENSE

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction

=====

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- o We don't promise that this software works. However, we will be interested in any kind of bug reports. (^ as is' distribution)
- o You can use this software for whatever you want, in parts or full form, without having to pay us. (^ royalty-free' usage)
- o You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. (^ credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

""""

Portions of this software are copyright <year> The FreeType

Project (www.freetype.org). All rights reserved.

""

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms

=====

0. Definitions

Throughout this license, the terms `package', `FreeType Project', and `FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the `FreeType Project', be they named as alpha, beta or final release.

`You' refers to the licensee, or person using the project, where `using' is a generic term including compiling the project's source code as well as linking it to form a `program' or `executable'. This program is referred to as `a program using the FreeType engine'.

This license applies to all files distributed in the original FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

THE FREETYPE PROJECT IS PROVIDED `AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- o Redistribution of source code must retain this license file ('FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.
- o Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: 'FreeType Project', 'FreeType Engine', 'FreeType library', or 'FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

o freetype@nongnu.org

Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution.

If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

o freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

<http://www.freetype.org>

GPL v2

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for

this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based

on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent

access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any

patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED

OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

```
#####
```

--- fthash.c and fthash.h are covered by the following notices ---

```
/*
 * Copyright 2000 Computing Research Labs, New Mexico State University
 * Copyright 2001-2015
 * Francesco Zappa Nardelli
 *
 * Permission is hereby granted, free of charge, to any person obtaining a
 * copy of this software and associated documentation files (the "Software"),
 * to deal in the Software without restriction, including without limitation
 * the rights to use, copy, modify, merge, publish, distribute, sublicense,
 * and/or sell copies of the Software, and to permit persons to whom the
 * Software is furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
```

*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL
* THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY
* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT
* OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
* THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

/*****

*
* This file is based on code from bdf.c,v 1.22 2000/03/16 20:08:50
*
* taken from Mark Leisher's xmbdfed package
*
*/

#####

--- FreeType 2 PSaux module is covered by the following notices ---

Copyright 2006-2014 Adobe Systems Incorporated.

This software, and all works of authorship, whether in source or object code form as indicated by the copyright notice(s) included herein (collectively, the "Work") is made available, and may only be used, modified, and distributed under the FreeType Project License, LICENSE.TXT. Additionally, subject to the terms and conditions of the FreeType Project License, each contributor to the Work hereby grants to any individual or legal entity exercising permissions granted by the FreeType Project License and this section (hereafter, "You" or "Your") a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such contributor that are necessarily infringed by their contribution(s) alone or by combination of their contribution(s) with the Work to which such contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

By using, modifying, or distributing the Work you indicate that you

have read and understood the terms and conditions of the FreeType Project License as well as those provided in this section, and you accept them fully.

#####

...

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Little Color Management System (LCMS) v2.12

LCMS License

<pre>

Little Color Management System
Copyright (c) 1998-2020 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

xml-commons/java/external/LICENSE.sax.txt \$Id: LICENSE.sax.txt 225954 2002-01-31 23:26:48Z curcuru \$

This license came from: <http://www.megginson.com/SAX/copying.html>

However please note future versions of SAX may be covered

under <http://saxproject.org/?selected=pd>

This page is now out of date -- see the new SAX site at

<http://www.saxproject.org/> for more up-to-date

releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain.

SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com

2000-05-05

ASM Bytecode Manipulation Framework v6.0

ASM License

<pre>

Copyright (c) 2000-2011 France Tlcom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The Unicode Standard, Unicode Character Database, Version 10.0.0

Unicode Character Database

...

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

Unicode Data Files do not include PDF online code charts under the directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard or under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

NOTICE TO USER: Carefully read the following legal agreement.
BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S
DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),
YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE
TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE
THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining
a copy of the Unicode data files and any associated documentation
(the "Data Files") or Unicode software and any associated documentation
(the "Software") to deal in the Data Files or Software
without restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, and/or sell copies of
the Data Files or Software, and to permit persons to whom the Data Files
or Software are furnished to do so, provided that either
(a) this copyright and permission notice appear with all copies
of the Data Files or Software, or
(b) this copyright and permission notice appear in associated
Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF
ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
NONINFRINGEMENT OF THIRD PARTY RIGHTS.
IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS
NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL
DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE,
DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER
TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR
PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder
shall not be used in advertising or otherwise to promote the sale,
use or other dealings in these Data Files or Software without prior
written authorization of the copyright holder.

...

Mozilla Elliptic Curve Cryptography (ECC)

Mozilla ECC Notice

This notice is provided with respect to Elliptic Curve Cryptography, which is included with JRE, JDK, and OpenJDK.

You are receiving a [copy](<http://hg.openjdk.java.net/jdk9/jdk9/jdk/file/tip/src/jdk.crypto.ec/share/native/libsunec/impl>) of the Elliptic Curve Cryptography library in source form with the JDK and OpenJDK source distributions, and as object code in the JRE & JDK runtimes.

<pre>

In the case of the JRE & JDK runtimes, the terms of the Oracle license do NOT apply to the Elliptic Curve Cryptography library; it is licensed under the following license, separately from Oracle's JDK & JRE. If you do not wish to install the Elliptic Curve Cryptography library, you may delete the Elliptic Curve Cryptography library:

- On Solaris and Linux systems: delete \$(JAVA_HOME)/lib/libsunec.so
- On Mac OSX systems: delete \$(JAVA_HOME)/lib/libsunec.dylib
- On Windows systems: delete \$(JAVA_HOME)\bin\sunec.dll

</pre>

Written Offer for Source Code

<pre>

For third party technology that you receive from Oracle in binary form which is licensed under an open source license that gives you the right to receive the source code for that binary, you can obtain a copy of the applicable source code from this page:

<http://hg.openjdk.java.net/jdk9/jdk9/jdk/file/tip/src/jdk.crypto.ec/share/native/libsunec/impl>

If the source code for the technology was not provided to you with the binary, you can also receive a copy of the source code on physical media by submitting a written request to:

Oracle America, Inc.
Attn: Associate General Counsel,
Development and Engineering Legal
500 Oracle Parkway, 10th Floor
Redwood Shores, CA 94065

Or, you may send an email to Oracle using the form at:

<http://www.oracle.com/goto/opensourcecode/request>

Your request should include:

- The name of the component or binary file(s) for which you are requesting the source code

- The name and version number of the Oracle product containing the binary
- The date you received the Oracle product
- Your name
- Your company name (if applicable)
- Your return mailing address and email and
- A telephone number in the event we need to reach you.

We may charge you a fee to cover the cost of physical media and processing. Your request must be sent (i) within three (3) years of the date you received the Oracle product that included the component or binary file(s) that are the subject of your request, or (ii) in the case of code licensed under the GPL v3, for as long as Oracle offers spare parts or customer support for that product model.

</pre>

LGPL 2.1

<pre>

GNU LESSER GENERAL PUBLIC LICENSE
Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better

strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of

this Lesser General Public License (also called "this License").

Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) The modified work must itself be a software library.

b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.

c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify

that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above

specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by

modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries,

so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

```
This library is free software; you can redistribute it and/or  
modify it under the terms of the GNU Lesser General Public  
License as published by the Free Software Foundation; either  
version 2.1 of the License, or (at your option) any later version.
```

```
This library is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU  
Lesser General Public License for more details.
```

```
You should have received a copy of the GNU Lesser General Public  
License along with this library; if not, write to the Free Software  
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
```

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the  
library `Frob' (a library for tweaking knobs) written by James Random Hacker.
```

```
<signature of Ty Coon>, 1 April 1990  
Ty Coon, President of Vice
```

That's all there is to it!

```
</pre>
```

```
## xwd v1.0.7
```

```
### xwd utility
```

```
<pre>
```

This is the copyright for the files in src/java.desktop/unix/native/libawt_xawt:
list.h, multiVis.h, wsutils.h, list.c, multiVis.c

Copyright 1994 Hewlett-Packard Co.

Copyright 1996, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

```
</pre>
```

Copyright (c) 2009, Orbitz World Wide

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the Orbitz World Wide nor the names of its contributors may be used to endorse or promote products derived from this software

without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache HttpClient Mime

Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

The project util-linux doesn't use the same license for all of the code.
There is code under:

- * GPLv3+ (GNU General Public License version 3, or any later version)
- * GPLv2+ (GNU General Public License version 2, or any later version)
- * GPLv2 (GNU General Public License version 2)
- * LGPLv2+ (GNU Lesser General Public License v2 (or 2.1) or any later version)
- * BSD with advertising
- * Public Domain

Please, check the source code for more details. A license is usually at the start of each source file.

The /COPYING file (GPLv2+) is the default license for code without an explicitly defined license.

```
## CUP Parser Generator for Java v 0.11b
```

```
### CUP Parser Generator License
```

```
<pre>
```

Copyright 1996-2015 by Scott Hudson, Frank Flannery, C. Scott Ananian, Michael Petter

Permission to use, copy, modify, and distribute this software and its

documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Santuario v2.2.1

Apache Santuario Notice

<pre>

Apache Santuario - XML Security for Java
Copyright 1999-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

It was originally based on software copyright (c) 2001, Institute for
Data Communications Systems, <<http://www.nue.et-inf.uni-siegen.de/>>.

The development of this software was partly funded by the European
Commission in the <WebSig> project in the ISIS Programme.

</pre>

Apache 2.0 License

<pre>

Apache License

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions

to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices

stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

</pre>

Harfbuzz v2.3.1

Harfbuzz License

<http://cgit.freedesktop.org/harfbuzz/tree/COPYING>

<pre>

HarfBuzz is licensed under the so-called "Old MIT" license. Details follow. For parts of HarfBuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright 2010,2011,2012 Google, Inc.
Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stribley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 2004,2007,2008,2009,2010 Red Hat, Inc.
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

All source code, except for one section, is licensed as above. The one exception is licensed with a slightly different MIT variant:

The contents of this directory are licensed under the following terms:

Copyright (C) 2012 Grigori Goronzy <greg@kinoho.net>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS MATH DERIVATIVE WORKS:

The Apache commons-math library includes a number of subcomponents whose implementation is derived from original sources written in C or Fortran. License terms of the original sources are reproduced below.

=====
For the lmdcr, lmpar and qrslv Fortran routine from minpack and translated in the LevenbergMarquardtOptimizer class in package

org.apache.commons.math.optimization.general

Original source copyright and license statement:

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF

ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

=====

Copyright and license statement for the odex Fortran routine developed by E. Hairer and G. Wanner and translated in GraggBulirschStoerIntegrator class in package org.apache.commons.math.ode.nonstiff:

Copyright (c) 2004, Ernst Hairer

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

Copyright and license statement for the original lapack fortran routines translated in EigenDecompositionImpl class in package org.apache.commons.math.linear:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

\$COPYRIGHT\$

Additional copyrights may follow

\$HEADERS\$

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.

- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

Copyright and license statement for the original Mersenne twister C routines translated in MersenneTwister class in package org.apache.commons.math.random:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the

documentation and/or other materials provided with the distribution.

3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Thai Dictionary

Thai Dictionary License

<pre>

Copyright (C) 1982 The Royal Institute, Thai Royal Government.

Copyright (C) 1998 National Electronics and Computer Technology Center,
National Science and Technology Development Agency,
Ministry of Science Technology and Environment,
Thai Royal Government.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

Apache HttpCore
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache Log4j API
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache Log4j Core
Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

ResolverUtil.java
Copyright 2005-2006 Tim Fennell

Apache HttpCore NIO
Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

ADDITIONAL INFORMATION ABOUT LICENSING

Certain files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception.

Note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0 and may include FreeType. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result.

Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further distribute the package.

Failing to distribute notices associated with some files may also create unexpected legal consequences.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0. To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

OASIS PKCS #11 Cryptographic Token Interface v3.0

OASIS PKCS #11 Cryptographic Token Interface License

<pre>

Copyright OASIS Open 2020. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [<http://www.oasis-open.org/policies-guidelines/ipr>]

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind,

provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an

attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

</pre>

PLEASE READ THIS NOTICE CAREFULLY

This AppDynamics PHP-agent uses third-party library ZeroMQ v3.2.5.

The above mentioned third-party library has been modified to function as per the agent requirement. The original source code can be found here: <http://download.zeromq.org/>.

International Components for Unicode (ICU4J) v60.2

ICU4J License

...

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

Unicode Data Files do not include PDF online code charts under the directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard or under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

NOTICE TO USER: Carefully read the following legal agreement.

BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.

IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

...

Cryptix AES v3.2.0

Cryptix General License

<pre>

Cryptix General License

Copyright (c) 1995-2005 The Cryptix Foundation Limited.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice,

this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE CRYPTIX FOUNDATION LIMITED AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CRYPTIX FOUNDATION LIMITED OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
</pre>
## JLine v3.14.0
```

```
### JLine License
<pre>
```

Copyright (c) 2002-2018, the original author or authors.
All rights reserved.

<https://opensource.org/licenses/BSD-3-Clause>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS

"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

JRuby Joni v2.1.16

MIT License

<pre>

Copyright (c) 2017 JRuby Team

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

GIFLIB v5.2.1

GIFLIB License

...

The GIFLIB distribution is Copyright (c) 1997 Eric S. Raymond

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

<https://sourceforge.net/p/giflib/code/ci/master/tree/openbsd-reallocarray.c>

Copyright (c) 2008 Otto Moerbeek <otto@drijf.net>

SPDX-License-Identifier: MIT

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free

software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could

make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its

content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified it, and giving a relevant date.
- b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
- c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
- d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user

actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating

where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically

receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a

patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a

covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT

HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

You should have received a copy of the GNU General Public License
along with this program. If not, see <<http://www.gnu.org/licenses/>>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short
notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>  
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate
parts of the General Public License. Of course, your program's commands
might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school,
if any, to sign a "copyright disclaimer" for the program, if necessary.
For more information on this, and how to apply and follow the GNU GPL, see
<<http://www.gnu.org/licenses/>>.

The GNU General Public License does not permit incorporating your program
into proprietary programs. If your program is a subroutine library, you
may consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License. But first, please read
<<http://www.gnu.org/philosophy/why-not-lgpl.html>>.

```
## libpng v1.6.37
```

```
### libpng License
```

```
<pre>
```

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE

=====

PNG Reference Library License version 2

- * Copyright (c) 1995-2019 The PNG Reference Library Authors.
- * Copyright (c) 2018-2019 Cosmin Truta.
- * Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson.

- * Copyright (c) 1996-1997 Andreas Dilger.
- * Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabudhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat

Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

</pre>

Apache Commons Logging
Copyright 2003-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache Log4j Commons Logging Bridge
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

xml-commons/java/external/README.sax.txt \$Id: README.sax.txt 225954 2002-01-31 23:26:48Z curcuru \$

HEAR YE, HEAR YE!

All of the .java software and associated documentation about SAX in this repository are distributed freely in the public domain.

LICENSE.sax.txt covers all software and documentation from the megginson.com including the following in the xml-commons project:

```
xml-commons/java/external/src/org/xml/sax
and all subdirectories
xml-commons/java/external/xdocs/sax
and all subdirectories
```

The actual SAX classes in xml-commons came from:

```
http://www.megginson.com/Software/index.html
The original versions are tagged 'SAX-2_0-r2-prerelease'
## Unicode Common Local Data Repository (CLDR) v33
```

```
### CLDR License
```

```
...
```

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories

```
http://www.unicode.org/Public/, http://www.unicode.org/reports/,
http://www.unicode.org/cldr/data/,
http://source.icu-project.org/repos/icu/, and
http://www.unicode.org/utility/trac/browser/.
```

Unicode Data Files do not include PDF online code charts under the directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard or under the directories

```
http://www.unicode.org/Public/, http://www.unicode.org/reports/,
http://www.unicode.org/cldr/data/,
http://source.icu-project.org/repos/icu/, and
http://www.unicode.org/utility/trac/browser/.
```

NOTICE TO USER: Carefully read the following legal agreement.

BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.
IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.
Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

...

=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====

Apache Xerces Java
Copyright 1999-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.

- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

The copyrights in this software and any visual or audio work distributed with the software belong to Azul Systems, Inc. and those included in all other notice files either listed in the readme file or contained in any other included notice files with this distribution. All rights are reserved. Installation of this software and any Azul software bundled with or derived from this software is licensed only in accordance with these terms.

Provided you have not received the software directly from Azul and have already agreed to the terms of a separate license agreement, by installing, using or distributing this software you, on your own behalf and on behalf of your employer or principal, agree to be bound by these terms. If you do not agree to any of these terms, you may not use, copy, transmit, distribute nor install this software.

The software is developed and owned by Azul and/or any of its affiliates, subsidiaries or respective suppliers and licensors. The software also includes certain software components and materials ("Open Source Materials") of third parties ("Third Party Licensors") licensed under certain licenses ("Open Source Licenses").

Nothing in this Agreement limits your rights under, or grants you rights that supersede the terms and conditions of any applicable Open Source Licenses. You must review the Open Source Licenses located at http://www.azulsystems.com/license/zulu_third_party_licenses.html to understand your rights under them. In the event that no such third party license agreements exist, the restrictions contained in this Agreement shall apply.

With respect to the Open Source Licenses the following shall apply:

- 1) You hereby acknowledge and agree that you will be licensing any Open Source Materials directly from the applicable Third Party Licensors including the right to use such Third Party Materials in connection with the software
- 2) In the event of any inconsistencies or conflicting provisions of the Open Source Licenses and the provisions of this Agreement, the provisions of the Open Source Licenses shall prevail.

THIS SOFTWARE IS PROVIDED BY AZUL "AS-IS" AND WITHOUT EXPRESS OR IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL AZUL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES (INCLUDING BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICE, LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION,

HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY WHETHER IN CONTRACT, STRICT LIABILITY OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Apache Commons Codec
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from <http://aspell.net/test/orig/batch0.tab>.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

=====

The content of package org.apache.commons.codec.language.bm has been translated
from the original php source code available at <http://stevemorse.org/phoneticinfo.htm>
with permission from the original authors.

Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache HttpClient
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).
xml-commons/java/external/LICENSE.dom-documentation.txt \$Id: LICENSE.dom-documentation.txt 226215
2005-06-03 22:49:13Z mrglavas \$

This license came from: <http://www.w3.org/Consortium/Legal/copyright-documents-20021231>

W3C DOCUMENT LICENSE
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>

Public documents on the W3C site are provided by the copyright holders under
the following license. By using and/or copying this document, or the W3C

document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [date-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle \$Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) \$

A Classpath Exception File means any source file contained in this distribution which contains the following words in such files header: Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code.

Azul Systems, Inc. hereby confirms that each Classpath Exception File is subject to the clarification and special exception to the GPL that is outlined in the accompanying LICENSE file (under the heading CLASSPATH EXCEPTION TO THE GPL).

Apache Commons FileUpload

Copyright 2002-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

1.159 lz4-and-xxhash 1.7.1

1.159.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.160 fabric8-:::-kubernetes-model-:::- autoscaling 4.10.3

1.160.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.161 javassist 3.27.0

1.161.1 Available under license :

Found license 'GNU Lesser General Public License' in '<CENTER>MOZILLA PUBLIC LICENSE
Version portions thereof) with or without Modifications, and/or as part of a Larger future version of this License released under Section 6.1, and You must ("Netscape") may publish revised and/or new versions of the License from published under a particular version of the License, You may always continue Covered Code under the terms of any subsequent version of the License and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public admission of liability.13. MULTIPLE-LICENSED CODE. "Multiple-Licensed". "Multiple-Licensed" means that the Initial <P>
EXHIBIT A -Mozilla Public License. The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <P>Software distributed under the License is distributed on an "AS IS" basis, Copyright (C) 1999- Shigeru Chiba. All Rights Reserved. terms of the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), or the Apache License Version 2.0 (the "AL"), use your version of this software under the terms of the MPL, indicate'

Found license 'GNU Lesser General Public License' in '* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved. * The contents of this file are subject to the Mozilla Public License Version * 1.1 (the "License"); you may not use this file except in compliance with * the License. Alternatively, the contents of this file may be used under * the terms of the GNU Lesser General Public License Version 2.1 or later, * or the Apache License Version 2.0. * Software distributed under the License is distributed on an "AS IS" basis,'

Found license 'GNU Lesser General Public License' in 'Copyright (C) 1999-2020 by Shigeru Chiba, All rights reserved. This software is distributed under the Mozilla Public License Version 1.1, the GNU Lesser General Public License Version 2.1 or later, or the Apache License Version 2.0.'

Found license 'GNU Lesser General Public License' in '* Copyright (C) 2004 Bill Burke. All Rights Reserved. * The contents of this file are subject to the Mozilla Public License Version * 1.1 (the "License"); you may not use this file except in compliance with * the License. Alternatively, the contents of this file may be used under * the terms of the GNU Lesser General Public License Version 2.1 or later, * or the Apache License Version 2.0. * Software distributed under the License is distributed on an "AS IS" basis,'

Found license 'GNU Lesser General Public License' in '<h3>Copyright (C) 1999-2020 by Shigeru Chiba, All rights reserved.</h3>
Copyright (C) 1999- Shigeru Chiba. All Rights Reserved. the Mozilla Public License Version 1.1 (the "License");
 you may not use this software except in compliance with the License. You may obtain a copy of the License at <p>Software distributed under the License is distributed on an "AS IS" Copyright (C) 1999- Shigeru Chiba. All Rights Reserved. terms of the GNU Lesser General Public License Version 2.1 or later (the "LGPL"), or the Apache License Version 2.0 (the "AL"), use your version of this software under the terms of the MPL, indicate'

Found license 'GNU Lesser General Public License' in '# This library is free software; you can redistribute it and/or # modify it under the terms of the GNU Lesser General Public # License as published by the Free Software Foundation; either # version 2.1 of the License, or any later version. # This library is distributed in the hope that it will be useful, # but WITHOUT ANY WARRANTY; without even the implied warranty of # MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU # Lesser General Public License for more details. # You should have received a copy of the GNU Lesser General Public' Found license 'GNU Lesser General Public License' in '* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved. * The contents of this file are subject to the Mozilla Public License Version * 1.1 (the "License"); you may not use this file except in compliance with * the License. Alternatively, the contents of this file may be used under * the terms of the GNU Lesser General Public License Version 2.1 or later, * or the Apache License Version 2.0. * Software distributed under the License is distributed on an "AS IS" basis, + " All Rights Reserved.";

1.162 kafka-avro-serializer 5.5.1

1.162.1 Available under license :

Confluent Community License Agreement
Version 1.0

This Confluent Community License Agreement Version 1.0 (the Agreement) sets forth the terms on which Confluent, Inc. (Confluent) makes available certain software made available by Confluent under this Agreement (the Software). BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO SUCH TERMS AND CONDITIONS, YOU MUST NOT USE THE SOFTWARE. IF YOU ARE RECEIVING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY. Licensee means you, an individual, or the entity on whose behalf you are receiving the Software.

1. LICENSE GRANT AND CONDITIONS.

1.1 License. Subject to the terms and conditions of this Agreement, Confluent hereby grants to Licensee a non-exclusive, royalty-free, worldwide, non-transferable, non-sublicenseable license during the term of this Agreement to: (a) use the Software; (b) prepare modifications and derivative works of the Software; (c) distribute the Software (including without limitation in source code or object code form); and (d) reproduce copies of the Software (the License). Licensee is not granted the right to, and Licensee shall not, exercise the License for an Excluded Purpose. For purposes of this Agreement, Excluded Purpose means making available any software-as-a-service, platform-as-a-service, infrastructure-as-a-service or other similar online service that competes with Confluent products or services that provide the Software.

1.2 Conditions. In consideration of the License, Licensees distribution of the Software is subject to the following conditions:

(a) Licensee must cause any Software modified by Licensee to carry prominent notices stating that Licensee modified the Software.

(b) On each Software copy, Licensee shall reproduce and not remove or alter all Confluent or third party copyright or other proprietary notices contained in the Software, and Licensee must provide the notice below with each copy.

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at <http://www.confluent.io/confluent-community-license>. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

1.3 Licensee Modifications. Licensee may add its own copyright notices to modifications made by Licensee and may provide additional or different license terms and conditions for use, reproduction, or distribution of Licensees modifications. While redistributing the Software or modifications thereof, Licensee may choose to offer, for a fee or free of charge, support, warranty, indemnity, or other obligations. Licensee, and not Confluent, will be responsible for any such obligations.

1.4 No Sublicensing. The License does not include the right to sublicense the Software, however, each recipient to which Licensee provides the Software may exercise the Licenses so long as such recipient agrees to the terms and conditions of this Agreement.

2. TERM AND TERMINATION. This Agreement will continue unless and until earlier terminated as set forth herein. If Licensee breaches any of its conditions or obligations under this Agreement, this Agreement will terminate automatically and the License will terminate automatically and permanently.

3. INTELLECTUAL PROPERTY. As between the parties, Confluent will retain all right, title, and interest in the Software, and all intellectual property rights therein. Confluent hereby reserves all rights not expressly granted to Licensee in this Agreement. Confluent hereby reserves all rights in its trademarks and service marks, and no licenses therein are granted in this Agreement.

4. DISCLAIMER. CONFLUENT HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE.

5. LIMITATION OF LIABILITY. CONFLUENT WILL NOT BE LIABLE FOR ANY DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, LOST PROFITS OR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, INDIRECT, OR DIRECT DAMAGES, HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THE FOREGOING SHALL APPLY TO THE EXTENT PERMITTED BY APPLICABLE LAW.

6.GENERAL.

6.1 Governing Law. This Agreement will be governed by and interpreted in accordance with the laws of the state of California, without reference to its conflict of laws principles. If Licensee is located within the United States, all disputes arising out of this Agreement are subject to the exclusive jurisdiction of courts located in Santa Clara County, California. USA. If Licensee is located outside of the United States, any dispute, controversy or claim arising out of or relating to this Agreement will be referred to and finally determined by arbitration in accordance with the JAMS International Arbitration Rules. The tribunal will consist of one arbitrator. The place of arbitration will be Palo Alto, California. The language to be used in the arbitral proceedings will be English. Judgment upon the award rendered by the arbitrator may be entered in any court having jurisdiction thereof.

6.2 Assignment. Licensee is not authorized to assign its rights under this Agreement to any third party. Confluent may freely assign its rights under this Agreement to any third party.

6.3 Other. This Agreement is the entire agreement between the parties regarding the subject matter hereof. No amendment or modification of this Agreement will be valid or binding upon the parties unless made in writing and signed by the duly authorized representatives of both parties. In the event that any provision, including without limitation any condition, of this Agreement is held to be unenforceable, this Agreement and all licenses and rights granted hereunder will immediately terminate. Waiver by Confluent of a breach of any provision of this Agreement or the failure by Confluent to exercise any right hereunder will not be construed as a waiver of any subsequent breach of that right or as a waiver of any other right.

Apache Kafka

Copyright 2016 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at <https://github.com/jersey/jersey/>.
Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported

commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses.

To find the details that apply to this artifact see the accompanying LICENSE file.

For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="Home page of The Apache Software Foundation">

<link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
<link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png">
<link rel="icon" type="image/png" href="/favicons/favicon-32x32.png" sizes="32x32">
<link rel="icon" type="image/png" href="/favicons/favicon-194x194.png" sizes="194x194">
<link rel="icon" type="image/png" href="/favicons/favicon-96x96.png" sizes="96x96">
<link rel="icon" type="image/png" href="/favicons/android-chrome-192x192.png" sizes="192x192">
```

```

<link rel="icon" type="image/png" href="/favicons/favicon-16x16.png" sizes="16x16">
<link rel="manifest" href="/favicons/manifest.json">
<link rel="shortcut icon" href="/favicons/favicon.ico">
<meta name="msapplication-TileColor" content="#603cba">
<meta name="msapplication-TileImage" content="/favicons/mstile-144x144.png">
<meta name="msapplication-config" content="/favicons/browserconfig.xml">
<meta name="theme-color" content="#303284">

<title>Apache License, Version 2.0</title>
<link href='https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700%7cDroid+Serif:400,700'
rel='stylesheet' type='text/css'>
<link href="/css/min.bootstrap.css" rel="stylesheet">
<link href="/css/styles.css" rel="stylesheet">

```

```

<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See
the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF
licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file
except in compliance with the License. You may obtain a copy of the License at .
http://www.apache.org/licenses/LICENSE-2.0 . Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions and limitations under the License. -->

```

```

</head>

```

```

<body>

```

```

<!-- Navigation -->

```

```

<header>

```

```

<nav class="navbar navbar-default navbar-fixed-top">

```

```

  <div class="container">

```

```

    <div class="navbar-header">

```

```

      <button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">

```

```

        <span class="sr-only">Toggle navigation</span>

```

```

        <span class="icon-bar"></span>

```

```

        <span class="icon-bar"></span>

```

```

        <span class="icon-bar"></span>

```

```

      </button>

```

```

      <a href="#" class="navbar-brand"><span class="glyphicon glyphicon-home"></span></a>

```

```

    </div>

```

```

    <div class="collapse navbar-collapse" id="mainnav-collapse">

```

```

      <div style="line-height:20px; padding-top:5px; float:left"><a href="/">Home</a>&nbsp;&raquo;&nbsp;&nbsp;<a
href="/licenses/">Licenses</a></div>

```

```

      <ul class="nav navbar-nav navbar-right">

```

```

        <li class="dropdown">

```

```

          <a href="#" class="dropdown-toggle" data-toggle="dropdown">About <span class="caret"></span></a>

```

```

          <ul class="dropdown-menu" role="menu">

```

```

            <li><a href="/foundation">Overview</a></li>

```

```

<li><a href="/foundation/members.html">Members</a></li>
<li><a href="/foundation/how-it-works.html">Process</a></li>
<li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
<li><a href="/foundation/glossary.html">Glossary</a></li>
<li><a href="/foundation/preFAQ.html">FAQ</a></li>
<li><a href="/foundation/contact.html ">Contact</a></li>
</ul>
</li>
<li><a href="/index.html#projects-list">Projects</a></li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">People <span class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
<li><a href="http://people.apache.org/">Overview</a></li>
<li><a href="http://people.apache.org/committer-index.html">Committers</a></li>
<li><a href="/foundation/how-it-works.html#meritocracy">Meritocracy</a></li>
<li><a href="/foundation/how-it-works.html#roles">Roles</a></li>
<li><a href="http://planet.apache.org/">Planet Apache</a></li>
  </ul>
</li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Get Involved <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
    <li><a href="/foundation/getinvolved.html">Overview</a></li>
<li><a href="http://community.apache.org/">Community Development</a></li>
    <li><a href="http://helpwanted.apache.org/">Help Wanted</a></li>
<li><a href="http://www.apachecon.com/">ApacheCon</a></li>
  </ul>
</li>
<li><a href="/dyn/closer.cgi">Download</a></li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Support Apache <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
<li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
<li><a href="/foundation/contributing.html">Donations</a></li>
<li><a href="/foundation/buy_stuff.html">Buy Stuff</a></li>
<li><a href="/foundation/thanks.html">Thanks</a></li>
  </ul>
</li>
</ul>
</div>
</div>
</nav>
</header>
<!-- / Navigation -->
<div class="container">
<div class="row">

```

```

<div class="col-md-9 col-sm-8 col-xs-12">
  
</div>
<div class="col-md-3 col-sm-4 col-xs-12">
  <div class="input-group" style="margin-bottom: 5px;">
    <script>
(function() {
var cx = '005703438322411770421:5mgshgrgx2u';
var gcse = document.createElement('script');
gcse.type = 'text/javascript';
gcse.async = true;
gcse.src = (document.location.protocol == 'https:' ? 'https:' : 'http:') +
  '//cse.google.com/cse.js?cx=' + cx;
var s = document.getElementsByTagName('script')[0];
s.parentNode.insertBefore(gcse, s);
})();
</script>
  <gcse:searchbox-only></gcse:searchbox-only>
</div>
  <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/governance/">The Apache Way</a>
  <a role="button" class="btn btn-block btn-default btn-xs"
href="https://community.apache.org/contributors/">Contribute</a>
  <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/thanks.html">ASF Sponsors</a>
</div>
</div>
<div class="container"><style type="text/css">
/* The following code is added by mdx_elementid.py
It was originally lifted from http://subversion.apache.org/style/site.css */
/*
* Hide class="elementid-permalink", except when an enclosing heading
* has the :hover property.
*/
.headerlink, .elementid-permalink {
visibility: hidden;
}
h2:hover > .headerlink, h3:hover > .headerlink, h1:hover > .headerlink, h6:hover > .headerlink, h4:hover >
.headerlink, h5:hover > .headerlink, dt:hover > .elementid-permalink { visibility: visible }</style>
<p>Apache License<br></br>Version 2.0, January 2004<br></br>
<a href="http://www.apache.org/licenses/">http://www.apache.org/licenses/</a> </p>
<p>TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION</p>
<p><strong><a name="definitions">1. Definitions</a></strong>.</p>
<p>"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.</p>
<p>"Licensor" shall mean the copyright owner or entity authorized by the
copyright owner that is granting the License.</p>
<p>"Legal Entity" shall mean the union of the acting entity and all other
entities that control, are controlled by, or are under common control with

```

that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a

perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

-

- You must give any other recipients of the Work or Derivative Works a copy of this License; and

- You must cause any modified files to carry prominent notices stating that You changed the files; and

- You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

- If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in

this License.

<p>5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.</p>

<p>6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.</p>

<p>7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.</p>

<p>8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.</p>

<p>9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.</p>

<p>END OF TERMS AND CONDITIONS</p>

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

```
Copyright [yyyy] [name of copyright owner]
```

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
</pre></div></div>
```

```
<!-- Footer -->
```

```
<footer class="bg-primary">
```

```
<div class="container">
```

```
<div class="row">
```

```
<br />
```

```
<div class="col-sm-1">
```

```
</div>
```

```
<div class="col-sm-2">
```

```
<h5 class="white">Community</h5>
```

```
<ul class="list-unstyled white" role="menu">
```

```
<li><a href="http://community.apache.org/">Overview</a></li>
```

```
<li><a href="/foundation/conferences.html">Conferences</a></li>
```

```
<li><a href="http://community.apache.org/gsoc.html">Summer of Code</a></li>
```

```
<li><a href="http://community.apache.org/newcomers/">Getting Started</a></li>
```

```
<li><a href="/foundation/how-it-works.html">The Apache Way</a></li>
```

```
<li><a href="/travel/">Travel Assistance</a></li>
```

```
<li><a href="/foundation/getinvolved.html">Get Involved</a></li>
```

```
<li><a href="http://community.apache.org/newbiefaq.html">Community FAQ</a></li>
```

```
</ul>
```

```
</div>
```

```
<div class="col-sm-2">
```

```
<h5 class="white">Innovation</h5>
  <ul class="list-unstyled white" role="menu">
    <li><a href="http://incubator.apache.org/">Incubator</a></li>
    <li><a href="http://labs.apache.org/">Labs</a></li>
    <li><a href="/licenses/">Licensing</a></li>
    <li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
    <li><a href="/foundation/marks/">Trademark Policy</a></li>
    <li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
  <h5 class="white">Tech Operations</h5>
  <ul class="list-unstyled white" role="menu">
    <li><a href="/dev/">Developer Information</a></li>
    <li><a href="/dev/infrastructure.html">Infrastructure</a></li>
    <li><a href="/security/">Security</a></li>
    <li><a href="http://status.apache.org">Status</a></li>
    <li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
  <h5 class="white">Press</h5>
  <ul class="list-unstyled white" role="menu">
    <li><a href="/press/">Overview</a></li>
    <li><a href="https://blogs.apache.org/">ASF News</a></li>
    <li><a href="https://blogs.apache.org/foundation/">Announcements</a></li>
    <li><a href="https://twitter.com/TheASF">Twitter Feed</a></li>
    <li><a href="/press/#contact">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
  <h5 class="white">Legal</h5>
  <ul class="list-unstyled white" role="menu">
    <li><a href="/legal/">Legal Affairs</a></li>
    <li><a href="/licenses/">Licenses</a></li>
    <li><a href="/foundation/marks/">Trademark Policy</a></li>
    <li><a href="/foundation/records/">Public Records</a></li>
    <li><a href="/foundation/policies/privacy.html">Privacy Policy</a></li>
    <li><a href="/licenses/exports/">Export Information</a></li>
    <li><a href="/foundation/license-faq.html">License/Distribution FAQ</a></li>
    <li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-1">
```

```
</div>

</div>
<hr class="col-lg-12 hr-white" />
<div class="row">
  <div class="col-lg-12">
    <p class="text-center">Copyright &#169; 2016 The Apache Software Foundation, Licensed under the <a
class="white" href="http://www.apache.org/licenses/LICENSE-2.0">Apache License, Version 2.0</a>.</p>
    <p class="text-center">Apache and the Apache feather logo are trademarks of The Apache Software
Foundation.</p>
  </div>
</div>
</div>

</footer>

<!-- / Footer -->

<script src="/js/jquery-2.1.1.min.js"></script>
<script src="/js/bootstrap.js"></script>
</body>
</html>
```

This copy of Jackson JSON processor annotations is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>
Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://opensource.org/licenses/mit-license.php">here</a>.</p>
</body></html>
```

Format: <http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/>

Upstream-Name: schema-registry

Source: <https://github.com/confluentinc/schema-registry>

Files: *

Copyright: 2018 Confluent, Inc.

License: Apache-2

Files: core/*

Copyright: 2015 Confluent, Inc.

License: Confluent Community License

License: Confluent Community License

Licensed under the Confluent Community License; you may not use this file except in compliance with the License. You may obtain a copy of the License at

.
<http://www.confluent.io/confluent-community-license>

.
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

License: Apache-2

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

.
<http://www.apache.org/licenses/LICENSE-2.0>

.
Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

.
On Debian systems, the Apache 2.0 license can be found in
</usr/share/common-licenses/Apache-2.0>.

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This distribution has a binary dependency on jersey, which is available under the CDDL License as described below.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL - Version 1.1)

1. Definitions.

- 1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
- 1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
- 1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
- 1.4. Executable means the Covered Software in any form other than Source Code.
- 1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
- 1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
- 1.7. License means this document.
- 1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
- 1.9. Modifications means the Source Code and Executable form of any of the following:
 - A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
 - B. Any new file that contains any part of the Original Software or previous Modification; or
 - C. Any new file that is contributed or otherwise made available under the terms of this License.
- 1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
- 1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or

a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible

for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Apache Avro

Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

The following commands were used to generate license and notice files. Replace <VERSION> with the Schema Registry version, <SRC_PATH> with the path to the Schema Registry source directory, and <LICENSE_TOOL_PATH> with the path of the license tool.

```
cd <SRC_PATH>
mvn package -DskipTests
mkdir /tmp/jars
mkdir /tmp/overrides
cp package-schema-registry/target/kafka-schema-registry-package-<VERSION>-package/share/java/schema-registry/*.jar /tmp/jars/
cp package-kafka-serde-tools/target/kafka-serde-tools-package-<VERSION>-package/share/java/kafka-serde-tools/*.jar /tmp/jars/
cd <LICENSE_TOOL_PATH>
./bin/run_license_job.bash -i /tmp/jars -l <SRC_PATH>/licenses -n <SRC_PATH>/notices -h
<SRC_PATH>/licenses-and-notices.html -o /tmp/overrides
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

<html lang="en-US">
<head>
<title>GlassFish Dual License Header and License Notice GPL v2 and CDDL</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

```
<style type="text/css">
li {display: block;}
</style>
</head>
```

```
<body leftmargin="0" topmargin="0" marginheight="0" marginwidth="0" bgcolor="#ffffff">
```

```
<h3>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0</h3>
```

```
<ul>
```

```
<li><p><b>1. Definitions.</b></p>
```

```
<ul>
```

```
<li>
```

```
<p>
```

```
<b>1.1. &ldquo;Contributor&rdquo;</b> means each individual or
entity that creates or contributes to the creation of
Modifications.
```

```
</p>
```

```
</li>
```

```
<li>
```

```
<p>
```

```
<b>1.2. &ldquo;Contributor Version&rdquo;</b> means
the combination of the Original Software, prior
Modifications used by a Contributor (if any), and the
Modifications made by that particular Contributor.
```

```
</p>
```

```
</li>
```

```
<li>
```

```
<p>
```

```
<b>1.3. &ldquo;Covered Software&rdquo;</b> means (a)
the Original Software, or (b) Modifications, or (c) the
combination of files containing Original Software with files
containing Modifications, in each case including portions
thereof.
```

```
</p>
```

```
</li>
```

```
<li>
```

```
<p>
```

```
<b>1.4. &ldquo;Executable&rdquo;</b> means the
Covered Software in any form other than Source Code.
```

```
</p>
```

```
</li>
```

```
<li>
```

```
<p>
```

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

<p>

C. Any new file that is contributed or otherwise made available under the terms of this License.

</p>

<p>

1.10. “Original Software” means the Source Code and Executable form of computer software code that is originally released under this License.

</p>

<p>

1.11. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

</p>

<p>

1.12. “Source Code” means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

</p>

<p>

1.13. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract

or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

</p>

<p>2. License Grants.</p>

<p>2.1. The Initial Developer Grant.</p>

<p>

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

</p>

<p>

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

</p>

<p>

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

</p>

<p>

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

</p>

<p>
(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

</p>

<p>2.2. Contributor Grant.</p>

<p>

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

</p>

<p>

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

</p>

<p>

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

</p>

<p>

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

</p>

<p>3.3. Required Notices.</p>

<p>

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

</p>

<p>3.4. Application of Additional Terms.</p>

<p>

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

</p>

<p>3.5. Distribution of Executable Versions.</p>

<p>

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code

form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

</p>

<p>3.6. Larger Works.</p>

<p>

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

</p>

<p>4. Versions of the License.</p>

<p>4.1. New Versions.</p>

<p>

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

</p>

<p>4.2. Effect of New Versions.</p>

<p>

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any

subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

</p>

<p>4.3. Modified Versions.</p>

<p>

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You:

- (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and
- (b) otherwise make it clear that the license contains terms which differ from this License.

</p>

<p>5. DISCLAIMER OF WARRANTY.</p>

<p>

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN “AS IS” BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

</p>

<p>6. TERMINATION.</p>

<p>

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

</p>

<p>

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

</p>

<p>

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

</p>

7. LIMITATION OF LIABILITY.

<p>

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF

SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

</p>

<p>8. U.S. GOVERNMENT END USERS.</p>

<p>

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48

C.F.R. § 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

</p>

<p>9. MISCELLANEOUS.</p>

<p>

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts

located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

</p>

<p>10. RESPONSIBILITY FOR CLAIMS.</p>

<p>

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

</p>

<p>

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

</p>

<p>

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions).

Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

</p>

<h3>The GNU General Public License (GPL) Version 2, June 1991</h3>

<p>
Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

</p>

<p>
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

</p>

<p>Preamble</p>

<p>

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
License is intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.
This General Public License applies to most of the Free Software
Foundation's software and to any other program whose authors commit
to using it. (Some other Free Software Foundation software is covered
by the GNU Library General Public License instead.) You can apply it
to your programs, too.

</p>

<p>

When we speak of free software, we are referring to freedom, not
price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and
charge for this service if you wish), that you receive source code
or can get it if you want it, that you can change the software or
use pieces of it in new free programs; and that you know you can
do these things.

</p>

<p>

To protect your rights, we need to make restrictions that forbid
anyone to deny you these rights or to ask you to surrender the
rights. These restrictions translate to certain responsibilities
for you if you distribute copies of the software, or if you modify it.

</p>

<p>

For example, if you distribute copies of such a program, whether
gratis or for a fee, you must give the recipients all the rights
that you have. You must make sure that they, too, receive or can get
the source code. And you must show them these terms so they know
their rights.

</p>

<p>

We protect your rights with two steps: (1) copyright the software,
and (2) offer you this license which gives you legal permission to
copy, distribute and/or modify the software.

</p>

<p>

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

</p>

<p>

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

</p>

<p>

The precise terms and conditions for copying, distribution and modification follow.

</p>

<p>TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION</p>

<ul style="margin-left:0; padding-left:0; border-left:0">

<p>

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

</p>

<p>

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

</p>

<p>

1. You may copy and distribute verbatim copies of the Program's

source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

</p>

<p>

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

</p>

<p>

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

</p>

<p>

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

</p>

<p>

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

</p>

<p>

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

</p>

<p>

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

</p>

<p>

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

</p>

<p>

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

</p>

<p>

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

</p>

<p>

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

</p>

<p>

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

</p>

<p>

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

</p>

<p>

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

</p>

<p>

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

</p>

<p>

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

</p>

<p>

5. You are not required to accept this License, since you have

not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

</p>

<p>

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

</p>

<p>

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

</p>

<p>

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

</p>

<p>

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that

choice.

</p>

<p>

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

</p>

<p>

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

</p>

<p>

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

</p>

<p>

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

</p>

<p>

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

</p>

<p>NO WARRANTY</p>

<p>
11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.
</p>

<p>
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.
</p>

<p>END OF TERMS AND CONDITIONS</p>

<p>How to Apply These Terms to Your New Programs</p>
<p>

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

</p>
<p>
To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a

pointer to where the full notice is found.

</p>

<p>

One line to give the program's name and a brief idea of what it does.

</p>

<p>

Copyright (C) <year> <name of author>

</p>

<p>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

</p>

<p>

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

</p>

<p>

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

</p>

<p>

Also add information on how to contact you by electronic and paper mail.

</p>

<p>

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

</p>

<p>

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

</p>

<p>

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

</p>

<p>

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

</p>

<p>

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

</p>

<p>

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

</p>

<p>

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

</p>

<li style="background-color:yellow;">

<p>"CLASSPATH" EXCEPTION TO THE GPL VERSION 2</p>

<p>

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

</p>

<p>
Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

</p>

<p>

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

</p>

</body>

</html>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />

<title>Untitled Document</title>

</head>

<body>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">

<HTML>

<!-- @page { margin: 0.79in } P { margin-bottom: 0.08in } H1 { margin-top: 0in; margin-bottom: 0.17in; widows: 0; orphans: 0 } H1.western { font-family: "Times New Roman", serif; font-size: 12pt; font-weight: normal } H1.cjk { font-family: "Andale Sans UI"; font-size: 12pt; font-weight: normal } H1.cnl { font-family: "Tahoma"; font-size: 12pt; font-weight: normal } -->

<BODY DIR="LTR">

<p>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1</p>

<p>1. Definitions.</p>

<blockquote>1.1. “Contributor” means each individual or entity that creates or contributes to the creation of Modifications.</blockquote>

<blockquote>1.2. “Contributor Version” means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.</blockquote>

<blockquote>1.3. “Covered Software” means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including

portions thereof.</blockquote>

<blockquote>1.4. “Executable” means the Covered Software in any form other than Source Code.</blockquote>

<blockquote>1.5. “Initial Developer” means the individual or entity that first makes Original Software available under this License.</blockquote>

<blockquote>1.6. “Larger Work” means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.</blockquote>

<blockquote>1.7. “License” means this document.</blockquote>

<blockquote>1.8. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.</blockquote>

<blockquote>1.9. “Modifications” means the Source Code and Executable form of any of the following:</blockquote>

<blockquote>A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;</blockquote>

<blockquote>B. Any new file that contains any part of the Original Software or previous Modification; or</blockquote>

<blockquote>C. Any new file that is contributed or otherwise made available under the terms of this License.</blockquote>

<blockquote>1.10. “Original Software” means the Source Code and Executable form of computer software code that is originally released under this License.</blockquote>

<blockquote>1.11. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.</blockquote>

<blockquote>1.12. “Source Code” means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</blockquote>

<blockquote>1.13. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</blockquote>

<p>2. License Grants.</p>

<blockquote>2.1. The Initial Developer Grant.</blockquote>

<blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:</blockquote>

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).</blockquote>

<blockquote>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.</blockquote>

<blockquote>(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii)

the combination of the Original Software with other software or devices.</blockquote>

<blockquote>2.2. Contributor Grant.</blockquote>

<blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:</blockquote>

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).</blockquote>

<blockquote>(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.</blockquote>

<blockquote>(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.</blockquote>

<p>3. Distribution Obligations.</p>

<blockquote>3.1. Availability of Source Code.</blockquote>

<blockquote>Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.</blockquote>

<blockquote>3.2. Modifications.</blockquote>

<blockquote>The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.</blockquote>

<blockquote>3.3. Required Notices.</blockquote>

<blockquote>You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.</blockquote>

<blockquote>3.4. Application of Additional Terms.</blockquote>

<blockquote>You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.</blockquote>

<blockquote>3.5. Distribution of Executable Versions.</blockquote>

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial

Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as “Participant”) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.</blockquote>

<blockquote>6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.</blockquote>

<blockquote>6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.</blockquote>

<p>7. LIMITATION OF LIABILITY.</p>

<blockquote>

<p>UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.</p>

</blockquote>

<p>8. U.S. GOVERNMENT END USERS.</p>

<blockquote>

<p>The Covered Software is a “commercial item,” as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of “commercial computer software” (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and “commercial computer software documentation” as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.</p>

</blockquote>

<p>9. MISCELLANEOUS.</p>

<blockquote>

<p>This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the

Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

</blockquote>

<p>10. RESPONSIBILITY FOR CLAIMS.</p>

<blockquote>

<p>As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.</p>

</blockquote>

<hr />

<p>NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)</p>

<p>The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California. </p>

<p>

</p>

<h1>The GNU General Public License (GPL) Version 2, June 1991</h1>

<p>Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA</p>

<p>Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.</p>

<p>Preamble</p>

<p>The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.</p>

<p>When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.</p>

<p>To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.</p>

<p>For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And

you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you

distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License.

If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF

DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program 'Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

-

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this

library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.</p>

<p>

</p>

<p>

</p>

</body>

</html>

This copy of Jackson JSON processor is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/>

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/" The project is licensed under the Confluent Community License, except for client libs, which is under the Apache 2.0 license.

See LICENSE file in each subfolder for detailed license agreement.

Apache log4j

Copyright 2007 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).
Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License

Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
<meta name="description" content="Home page of The Apache Software Foundation">

<link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
<link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png">
<link rel="icon" type="image/png" href="/favicons/favicon-32x32.png" sizes="32x32">
<link rel="icon" type="image/png" href="/favicons/favicon-194x194.png" sizes="194x194">
<link rel="icon" type="image/png" href="/favicons/favicon-96x96.png" sizes="96x96">
<link rel="icon" type="image/png" href="/favicons/android-chrome-192x192.png" sizes="192x192">
<link rel="icon" type="image/png" href="/favicons/favicon-16x16.png" sizes="16x16">
<link rel="manifest" href="/favicons/manifest.json">
<link rel="shortcut icon" href="/favicons/favicon.ico">
<meta name="msapplication-TileColor" content="#603cba">
<meta name="msapplication-TileImage" content="/favicons/mstile-144x144.png">
<meta name="msapplication-config" content="/favicons/browserconfig.xml">
```

```
<meta name="theme-color" content="#303284">
```

```
<title>Licenses</title>
```

```
<link href='https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700%7cDroid+Serif:400,700'  
rel='stylesheet' type='text/css'>
```

```
<link href="/css/min.bootstrap.css" rel="stylesheet">
```

```
<link href="/css/styles.css" rel="stylesheet">
```

```
<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See  
the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF  
licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file  
except in compliance with the License. You may obtain a copy of the License at  
http://www.apache.org/licenses/LICENSE-2.0 . Unless required by applicable law or agreed to in writing, software  
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR  
CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing  
permissions and limitations under the License. -->
```

```
</head>
```

```
<body>
```

```
<!-- Navigation -->
```

```
<header>
```

```
<nav class="navbar navbar-default navbar-fixed-top">
```

```
<div class="container">
```

```
<div class="navbar-header">
```

```
<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">
```

```
<span class="sr-only">Toggle navigation</span>
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
<span class="icon-bar"></span>
```

```
</button>
```

```
<a href="#" class="navbar-brand"><span class="glyphicon glyphicon-home"></span></a>
```

```
</div>
```

```
<div class="collapse navbar-collapse" id="mainnav-collapse">
```

```
<div style="line-height:20px; padding-top:5px; float:left"><a href="/">Home</a>&nbsp;&raquo;&nbsp;&lt;a  
href="/licenses/">Licenses</a></div>
```

```
<ul class="nav navbar-nav navbar-right">
```

```
<li class="dropdown">
```

```
<a href="#" class="dropdown-toggle" data-toggle="dropdown">About <span class="caret"></span></a>
```

```
<ul class="dropdown-menu" role="menu">
```

```
<li><a href="/foundation">Overview</a></li>
```

```
<li><a href="/foundation/members.html">Members</a></li>
```

```
<li><a href="/foundation/how-it-works.html">Process</a></li>
```

```
<li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
```

```
<li><a href="/foundation/glossary.html">Glossary</a></li>
```

```
<li><a href="/foundation/preFAQ.html">FAQ</a></li>
```

```
<li><a href="/foundation/contact.html ">Contact</a></li>
```

```

    </ul>
  </li>
</li><a href="/index.html#projects-list">Projects</a></li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">People <span class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
<li><a href="http://people.apache.org/">Overview</a></li>
<li><a href="http://people.apache.org/committer-index.html">Committers</a></li>
<li><a href="/foundation/how-it-works.html#meritocracy">Meritocracy</a></li>
<li><a href="/foundation/how-it-works.html#roles">Roles</a></li>
<li><a href="http://planet.apache.org/">Planet Apache</a></li>
  </ul>
</li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Get Involved <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
<li><a href="/foundation/getinvolved.html">Overview</a></li>
<li><a href="http://community.apache.org/">Community Development</a></li>
  <li><a href="http://helpwanted.apache.org/">Help Wanted</a></li>
<li><a href="http://www.apachecon.com/">ApacheCon</a></li>
  </ul>
</li>
<li><a href="/dyn/closer.cgi">Download</a></li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Support Apache <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
<li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
<li><a href="/foundation/contributing.html">Donations</a></li>
<li><a href="/foundation/buy_stuff.html">Buy Stuff</a></li>
<li><a href="/foundation/thanks.html">Thanks</a></li>
  </ul>
</li>
</ul>
</div>
</nav>
</header>
<!-- / Navigation -->
<div class="container">
<div class="row">
<div class="col-md-9 col-sm-8 col-xs-12">
  
</div>
<div class="col-md-3 col-sm-4 col-xs-12">
  <div class="input-group" style="margin-bottom: 5px;">
<script>

```

```

(function() {
  var cx = '005703438322411770421:5mgshgrgx2u';
  var gcse = document.createElement('script');
  gcse.type = 'text/javascript';
  gcse.async = true;
  gcse.src = (document.location.protocol == 'https:' ? 'https:' : 'http:') +
    '//cse.google.com/cse.js?cx=' + cx;
  var s = document.getElementsByTagName('script')[0];
  s.parentNode.insertBefore(gcse, s);
})();
</script>
  <gcse:searchbox-only></gcse:searchbox-only>
</div>
  <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/governance/">The Apache Way</a>
  <a role="button" class="btn btn-block btn-default btn-xs"
href="https://community.apache.org/contributors/">Contribute</a>
  <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/thanks.html">ASF Sponsors</a>
</div>
</div>
</div>
<div class="container"><style type="text/css">
/* The following code is added by mdx_elementid.py
  It was originally lifted from http://subversion.apache.org/style/site.css */
/*
* Hide class="elementid-permalink", except when an enclosing heading
* has the :hover property.
*/
.headerlink, .elementid-permalink {
  visibility: hidden;
}
h2:hover > .headerlink, h3:hover > .headerlink, h1:hover > .headerlink, h6:hover > .headerlink, h4:hover >
.headerlink, h5:hover > .headerlink, dt:hover > .elementid-permalink { visibility: visible }</style>
<p>The Apache Software Foundation uses various licenses to <a href="#distributions">distribute
software and documentation</a>, to accept regular
<a href="#clas">contributions from individuals and corporations</a>, and to accept
larger <a href="#grants">grants of existing software products</a>.</p>
<p>These licenses help us achieve our goal of providing reliable and
long-lived software products through collaborative open source software
development. In all cases, contributors retain full rights to use their
original contributions for any other purpose outside of Apache while
providing the ASF and its projects the right to distribute and build upon
their work within Apache.</p>
<h1 id="distributions">Licensing of Distributions<a class="headerlink" href="#distributions" title="Permanent
link">&para;</a></h1>
<p>All software produced by The Apache Software Foundation or any of its
projects or subjects is licensed according to the terms of the documents
listed below.</p>
<h3 id="2.0">Apache License, Version 2.0 (current)<a class="headerlink" href="#2.0" title="Permanent

```

link">¶</h3>

<p>http://www.apache.org/licenses/LICENSE-2.0 (TXT or HTML)</p>

<p>The 2.0 version of the Apache License was approved by the ASF in 2004. The goals of this license revision have been to reduce the number of frequently asked questions, to allow the license to be reusable without modification by any project (including non-ASF projects), to allow the license to be included by reference instead of listed in every file, to clarify the license on submission of contributions, to require a patent license on contributions that necessarily infringe the contributor's own patents, and to move comments regarding Apache and other inherited attribution notices to a location outside the license terms (the NOTICE file).</p>

<p>The result is a license that is supposed to be compatible with other open source licenses, while remaining true to the original goals of the Apache Group and supportive of collaborative development across both nonprofit and commercial organizations. The Apache Software Foundation is still trying to determine if this version of the Apache License is compatible with the GPL.</p>

<p>All packages produced by the ASF are implicitly licensed under the Apache License, Version 2.0, unless otherwise explicitly stated. More developer documentation on how to apply the Apache License to your work can be found in * Applying the Apache License, Version 2.0 *.</p>

<h3 id="1.1">Apache License, Version 1.1 (historic)¶</h3>

<p>http://www.apache.org/licenses/LICENSE-1.1 </p>

<p>The 1.1 version of the Apache License was approved by the ASF in 2000. The primary change from the 1.0 license is in the 'advertising clause' (section 3 of the 1.0 license); derived products are no longer required to include attribution in their advertising materials, only in their documentation.</p>

<p>Individual packages licensed under the 1.1 version may have used different wording due to varying requirements for attribution or mark identification, but the binding terms were all the same.</p>

<h3 id="1.0">Apache License, Version 1.0 (historic)¶</h3>

<p>http://www.apache.org/licenses/LICENSE-1.0 </p>

<p>This is the original Apache License which applies only to older versions of Apache packages (such as version 1.2 of the Web server).</p>

<h1 id="clas">Contributor License Agreements¶</h1>

<p>The ASF desires that all contributors of ideas, code, or documentation to any Apache projects complete, sign, and submit (via fax or email) an Individual Contributor License Agreement (ICLA).

The purpose of this agreement is to clearly define the terms under which intellectual property has been contributed to the ASF and thereby allow us to defend the project should there be a legal dispute regarding the software at some future time. A signed ICLA is required to be

on file before an individual is given commit rights to an ASF project.</p>

<p>For a corporation that has assigned employees to work on an Apache project, a Corporate CLA (CCLA) is available for contributing intellectual property via the corporation, that may have been assigned as part of an employment agreement. Note that a Corporate CLA does not remove the need for every developer to sign their own ICLA as an individual, to cover any of their contributions which are not owned by the corporation signing the CCLA.</p>

<p>The ICLA is not tied to any employer you may have, so it is recommended to use one's personal email address in the contact details, rather than an @work address.</p>

<p>Your Full name will be published unless you provide an alternative Public name. For example if your full name is Andrew Bernard Charles Dickens, but you wish to be known as Andrew Dickens, please enter the latter as your Public name.</p>

<p>The email address and other contact details are not published.</p>

<h1 id="grants">Software Grants¶</h1>

<p>When an individual or corporation decides to donate a body of existing software or documentation to one of the Apache projects, they need to execute a formal Software Grant Agreement (SGA) with the ASF. Typically, this is done after negotiating approval with the ASF Incubator or one of the PMCs, since the ASF will not accept software unless there is a viable community available to support a collaborative project.</p>

<h1 id="submitting">Submitting License Agreements and Grants¶</h1>

<p>Documents may be submitted by fax or email.</p>

<p>If submitting by fax, please print, sign, and fax all pages of the document to +1-919-573-9199. Please send documents right-side-up, first page first; and send only one document per fax.</p>

<p>If submitting by email, please fill the form with a pdf viewer, print, sign, scan all pages into a single pdf file, and email the pdf file as an attachment to secretary@apache.org. If possible, send the attachment from the email address in the document.

Please send only one document per email.</p>

<p>If you prefer to sign electronically, please fill the form, save it locally (e.g. icla.pdf), and sign the file by preparing a detached PGP signature. For example,</p>

<blockquote>

<p>gpg --armor --detach-sign icla.pdf</p>

</blockquote>

<p>The above will create a file icla.pdf.asc. Send both the file and signature as attachments in the same email to secretary@apache.org. Please send only one document (file plus signature) per email. Please do not submit your public key to Apache. Instead, please upload your public key to pgpkeys.mit.edu. </p>

<p>The files typically are named
icla.pdf and icla.pdf.asc for individual agreements;
ccla.pdf and ccla.pdf.asc for corporate agreements;
software-grant.pdf and software-grant.pdf.asc for grants. </p>

<h1 id="crypto">Export restrictions¶</h1>

<p>For export restriction information, please consult our ASF Export Classifications page.</p>

<h1 id="trademarks">Trademark and Logo Usage¶</h1>

<p>For ASF trademark and logo usage information, please consult our ASF Trademark Use Policy page.</p>

<h1 id="questions">Questions?¶</h1>

<p>For answers to frequently asked licensing questions, please consult our Licensing Frequently Asked Questions page.</p></div>

<!-- Footer -->

<footer class="bg-primary">

<div class="container">

<div class="row">

<div class="col-sm-1">

</div>

<div class="col-sm-2">

<h5 class="white">Community</h5>

<ul class="list-unstyled white" role="menu">

Overview

Conferences

Summer of Code

Getting Started

The Apache Way

Travel Assistance

Get Involved

Community FAQ

</div>

<div class="col-sm-2">

<h5 class="white">Innovation</h5>

<ul class="list-unstyled white" role="menu">

Incubator

Labs

Licensing

Licensing FAQ

Trademark Policy

Contacts

</div>

<div class="col-sm-2">

<h5 class="white">Tech Operations</h5>

<ul class="list-unstyled white" role="menu">


```
<li><a href="/dev/">Developer Information</a></li>
<li><a href="/dev/infrastructure.html">Infrastructure</a></li>
<li><a href="/security/">Security</a></li>
<li><a href="http://status.apache.org">Status</a></li>
<li><a href="/foundation/contact.html">Contacts</a></li>
</ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Press</h5>
<ul class="list-unstyled white" role="menu">
<li><a href="/press/">Overview</a></li>
<li><a href="https://blogs.apache.org/">ASF News</a></li>
<li><a href="https://blogs.apache.org/foundation/">Announcements</a></li>
<li><a href="https://twitter.com/TheASF">Twitter Feed</a></li>
<li><a href="/press/#contact">Contacts</a></li>
</ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Legal</h5>
<ul class="list-unstyled white" role="menu">
<li><a href="/legal/">Legal Affairs</a></li>
<li><a href="/licenses/">Licenses</a></li>
<li><a href="/foundation/marks/">Trademark Policy</a></li>
<li><a href="/foundation/records/">Public Records</a></li>
<li><a href="/foundation/policies/privacy.html">Privacy Policy</a></li>
<li><a href="/licenses/exports/">Export Information</a></li>
<li><a href="/foundation/license-faq.html">License/Distribution FAQ</a></li>
<li><a href="/foundation/contact.html">Contacts</a></li>
</ul>
</div>
```

```
<div class="col-sm-1">
</div>
```

```
</div>
<hr class="col-lg-12 hr-white" />
<div class="row">
<div class="col-lg-12">
<p class="text-center">Copyright © 1999-2016 The Apache Software Foundation, Licensed under the <a
class="white" href="http://www.apache.org/licenses/LICENSE-2.0">Apache License, Version 2.0</a>.</p>
<p class="text-center">Apache and the Apache feather logo are trademarks of The Apache Software
Foundation.</p>
</div>
</div>
</div>
```

</footer>

<!-- / Footer -->

<script src="/js/jquery-2.1.1.min.js"></script>

<script src="/js/bootstrap.js"></script>

</body>

</html>

This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.

For additional credits (generally to people who reported problems) see CREDITS file.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.163 protocol-buffers-[bom] 3.17.2

1.163.1 Available under license :

This is free and unencumbered software released into the public domain.

Anyone is free to copy, modify, publish, use, compile, sell, or distribute this software, either in source code form or as a compiled binary, for any purpose, commercial or non-commercial, and by any means.

In jurisdictions that recognize copyright laws, the author or authors of this software dedicate any and all copyright interest in the software to the public domain. We make this dedication for the benefit of the public at large and to the detriment of our heirs and successors. We intend this dedication to be an overt act of relinquishment in perpetuity of all present and future rights to this software under copyright law.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.

IN NO EVENT SHALL THE AUTHORS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

For more information, please refer to <http://unlicense.org/>

Copyright 2008 Google Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license. This file contains a list of people who have made large contributions to the public version of Protocol Buffers.

Original Protocol Buffers design and implementation:

Sanjay Ghemawat <sanjay@google.com>

Jeff Dean <jeff@google.com>

Daniel Dulitz <daniel@google.com>

Craig Silverstein

Paul Haahr <haahr@google.com>

Corey Anderson <corin@google.com>

(and many others)

Proto2 C++ and Java primary author:

Kenton Varda <kenton@google.com>

Proto2 Python primary authors:

Will Robinson <robinson@google.com>

Petar Petrov <petar@google.com>

Java Nano primary authors:

Brian Duff <bduff@google.com>

Tom Chao <chaot@google.com>

Max Cai <maxtroy@google.com>

Ulas Kirazci <ulas@google.com>

Large code contributions:

Jason Hsueh <jasonh@google.com>

Joseph Schorr <jschorr@google.com>

Wenbo Zhu <wenboz@google.com>

Large quantity of code reviews:

Scott Bruce <sbruce@google.com>

Frank Yellin

Neal Norwitz <nnorwitz@google.com>

Jeffrey Yasskin <jyasskin@google.com>

Ambrose Feinstein <ambrose@google.com>

Documentation:

Lisa Carey <lcarey@google.com>

Maven packaging:

Gregory Kick <gak@google.com>

Patch contributors:

Kevin Ko <kevin.s.ko@gmail.com>

* Small patch to handle trailing slashes in --proto_path flag.

Johan Euphrosine <proppy@aminche.com>

* Small patch to fix Python CallMethod().

Ulrich Kunitz <kune@deine-taler.de>

* Small optimizations to Python serialization.

Leandro Lucarella <llucax@gmail.com>

* VI syntax highlighting tweaks.

* Fix compiler to not make output executable.

Dilip Joseph <dilip.antony.joseph@gmail.com>

* Heuristic detection of sub-messages when printing unknown fields in text format.

Brian Atkinson <nairb774@gmail.com>

* Added @Override annotation to generated Java code where appropriate.

Vincent Choinire <Choiniere.Vincent@hydro.qc.ca>
* Tru64 support.

Monty Taylor <monty.taylor@gmail.com>
* Solaris 10 + Sun Studio fixes.

Alek Storm <alek.storm@gmail.com>
* Slicing support for repeated scalar fields for the Python API.

Oleg Smolsky <oleg.smolsky@gmail.com>
* MS Visual Studio error format option.
* Detect unordered_map in stl_hash.m4.

Brian Olson <brianolson@google.com>
* gzip/zlib I/O support.

Michael Poole <mdpooles@troilus.org>
* Fixed warnings about generated constructors not explicitly initializing all fields (only present with certain compiler settings).
* Added generation of field number constants.

Wink Saville <wink@google.com>
* Fixed initialization ordering problem in logging code.

Will Pierce <willp@nuclei.com>
* Small patch improving performance of in Python serialization.

Alexandre Vassalotti <alexandre@peadrop.com>
* Emacs mode for Protocol Buffers (editors/protobuf-mode.el).

Scott Stafford <scott.stafford@gmail.com>
* Added Swap(), SwapElements(), and RemoveLast() to Reflection interface.

Alexander Melnikov <alm@sibmail.ru>
* HPUX support.

Oliver Jowett <oliver.jowett@gmail.com>
* Detect whether zlib is new enough in configure script.
* Fixes for Solaris 10 32/64-bit confusion.

Evan Jones <evanj@mit.edu>
* Optimize Java serialization code when writing a small message to a stream.
* Optimize Java serialization of strings so that UTF-8 encoding happens only once per string per serialization call.
* Clean up some Java warnings.
* Fix bug with permanent callbacks that delete themselves when run.

Michael Kucharski <m.kucharski@gmail.com>
* Added CodedInputStream.getTotalBytesRead().

Kacper Kowalik <xarthisius.kk@gmail.com>
* Fixed m4/acx_pthread.m4 problem for some Linux distributions.

William Orr <will@worrbase.com>
* Fixed detection of sched_yield on Solaris.
* Added atomicops for Solaris

Andrew Paprocki <andrew@ishiboo.com>
* Fixed minor IBM xLC compiler build issues
* Added atomicops for AIX (POWER)

1.164 jetbrains-kotlin-kotlin-stdlib-jdk7 1.3.71

1.164.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2017 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1162863430_1649200705.41/0/kotlin-stdlib-jdk7-1-3-71-sources-jar/kotlin/AutoCloseable.kt
* /opt/cola/permits/1162863430_1649200705.41/0/kotlin-stdlib-jdk7-1-3-71-sources-
jar/kotlin/internal/jdk7/JDK7PlatformImplementations.kt
```

1.165 micronaut-mongodb 4.1.0

1.165.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/StringToServerAddressConverter.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/package-info.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/StringToReadConcernConverter.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/MongoConverterRegistrar.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/StringToWriteConcernConverter.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/MongoSettings.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/test/AbstractMongoProcessFactory.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/DefaultMongoConfiguration.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/NamedMongoConfiguration.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/AbstractMongoConfiguration.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/test/MongoProcessFactory.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/DefaultMongoClientSettingsFactory.java
- * /opt/cola/permits/1331473860_1653514577.0135384/0/micronaut-mongo-core-4-1-0-sources-jar/io/micronaut/configuration/mongo/core/convert/StringToReadPreferenceConverter.java

1.166 io-grpc-grpc-protobuf 1.39.0

1.166.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/package-info.java

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/StatusProto.java

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoMethodDescriptorSupplier.java

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoServiceDescriptorSupplier.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoUtils.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1183891218_1655096184.4932175/0/grpc-protobuf-1-39-0-sources-1-jar/io/grpc/protobuf/ProtoFileDescriptorSupplier.java

1.167 junit-jupiter-junit-jupiter-api 5.7.1

1.167.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or

subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)*** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

- * **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
- * **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
- * **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
- * **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- * **a)*** it must be made available under this Agreement, or if the Program *(i)* is combined with other material in a separate file or files made available under a Secondary License, and *(ii)* the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- * **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual

property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.168 fabric8-::-kubernetes-model-::-apps

4.10.3

1.168.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.169 utils 6.2.0

1.169.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.170 micronaut-problem-json 2.2.2

1.170.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2021 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/conf/ProblemConfigurationProperties.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/ProblemErrorResponseProcessor.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/violations/Violation.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/HttpStatusType.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/conf/MixinThrowableProblem.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-  
jar/io/micronaut/problem/conf/ProblemConfiguration.java  
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
```

jar/io/micronaut/problem/conf/ThrowableProblemObjectMapperBeanCreatedEventListener.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/serde/ProblemSerdeImport.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/violations/ConstraintViolationThrowableProblem.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/ThrowableProblemHandler.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/violations/ProblemConstraintViolationExceptionHandler.java
No license file was found, but licenses were detected in source scan.

```
/*  
 * Copyright 2017-2020 original authors  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License");  
 * you may not use this file except in compliance with the License.  
 * You may obtain a copy of the License at  
 *  
 * https://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an "AS IS" BASIS,  
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 * See the License for the specific language governing permissions and  
 * limitations under the License.  
 */
```

Found in path(s):

* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/conf/package-info.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/package-info.java
* /opt/cola/permits/1331473572_1653068678.1544204/0/micronaut-problem-json-2-2-2-sources-
jar/io/micronaut/problem/violations/package-info.java

1.171 jackson-datatype-joda 2.13.2

1.171.1 Available under license :

This copy of Jackson JSON processor streaming parser/generator is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.172 kotlin-stdlib 1.6.10

1.172.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2016 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/ArrayIntrinsics.kt
* /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-
jar/kotlin/reflect/KVisibility.kt
* /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-
jar/kotlin/internal/InternalAnnotations.kt
* /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/String.kt
* /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Library.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/reflect/KDeclarationContainer.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Comparable.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Boolean.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Number.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Throwable.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/CharSequence.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/annotation/Annotations.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Function.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Any.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Nothing.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Unit.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/reflect/KParameter.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Iterator.kt
- * /opt/cola/permits/1331473828_1653068625.3355322/0/kotlin-stdlib-1-6-10-sources-jar/kotlin/Char.kt

1.173 apache-commons-lang 2.6

1.173.1 Available under license :

Apache Commons Lang (for Apache Directory Studio)
Copyright 2003-2012 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.174 rfc3986-uri 0.8

1.174.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 dmfs GmbH
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/FluentParameterList.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Query.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/encoding/Normalized.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/paths/Resolved.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/adapters/MultiParameter.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/parametertypes/BasicParameterType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/parametersets/Removing.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/validation/CharSets.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/authorities/Text.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/valuetypes/BooleanValueType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/encoding/Utils/FormPercentEncodingOutputStream.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/parametersets/Fluent.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/encoding/Encoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/encoding/XWwwFormUrlEncoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/authorities/StructuredAuthority.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/parameters/parametersets/Appending.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-

jar/org/dmfs/rfc3986/parameters/ValueType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/fragments/SimpleFragment.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/RelativeUri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/Resolved.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/StructuredPath.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/valuetypes/TextValueType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/queries/SimpleQuery.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/encoding/IdempotentEncoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/UriEncoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/encoding/Precoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/adapters/XwfueParameterList.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/OpaqueUri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Authority.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/adapters/TextParameter.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/LazyUri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/utils/Parsed.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/EmptyUri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/authorities/OptionalLazyAuthority.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/authorities/EncodedAuthority.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Path.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/LazyPath.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Scheme.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/paths/Text.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/encoding/FormEncoded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/EncodedPath.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/parametersets/EmptyParameterList.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/parametersets/Replacing.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-

jar/org/dmfs/rfc3986/schemes/Schemes.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/validation/BitMapCharSet.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/ParameterList.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/authorities/OptionalLazyUserInfo.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/encoding/FormPreceded.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/parameters/UrlEncodedParameter.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/EmptyPath.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/validation/Validated.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/ParameterType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/Extended.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/fragments/OptionalLazyFragment.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/Parameter.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/schemes/StringScheme.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/encoding/utils/PercentEncodingOutputStream.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/valuetypes/IntegerValueType.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Fragment.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/utils/Split.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/validation/CharSet.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/parametersets/BasicParameterList.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/StructuredUri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/queries/OptionalLazyQuery.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/schemes/OptionalLazyScheme.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/uris/Normalized.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-jar/org/dmfs/rfc3986/Uri.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/paths/Normalized.java
* /opt/cola/permits/1162862871_1620923716.86/0/rfc3986-uri-0-8-sources-1-
jar/org/dmfs/rfc3986/parameters/adapters/OptionalParameter.java

1.175 metrics-jetty 4.1.17

1.175.1 Available under license :

Apache-2.0

1.176 json-p 1.0.4

1.176.1 Available under license :

Found license 'General Public License 2.0' in '* Copyright (c) 2011-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in '* Copyright (c) 2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in '* Copyright (c) 2012-2013 Oracle and/or its affiliates. All rights reserved. * The contents of this file are subject to the terms of either the GNU * General Public License Version 2 only ("GPL") or the Common Development * and Distribution License("CDDL") (collectively, the "License"). You * may not use this file except in compliance with the License. You can * Oracle designates this particular file as subject to the "Classpath" * exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'General Public License 2.0' in 'Copyright (c) 2011-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

Found license 'Common Development and Distribution License (CDDL) 1.1' in 'Copyright (c) 2011-2013 Oracle and/or its affiliates. All rights reserved. The contents of this file are subject to the terms of either the GNU General Public License Version 2 only ("GPL") or the Common Development and Distribution License("CDDL") (collectively, the "License"). You may not use this file except in compliance with the License. You can Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the GPL Version 2 section of the License'

1.177 kafka-protobuf-provider 6.2.0

1.177.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```

<modelVersion>4.0.0</modelVersion>

<parent>
  <groupId>io.confluent</groupId>
  <artifactId>kafka-schema-registry-parent</artifactId>
  <version>6.2.0</version>
</parent>

<licenses>
  <license>
    <name>Confluent Community License</name>
    <url>http://www.confluent.io/confluent-community-license</url>
    <distribution>repo</distribution>
  </license>
  <license>
    <name>Apache License 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
    <distribution>repo</distribution>
  </license>
</licenses>

<artifactId>kafka-protobuf-provider</artifactId>
<packaging>jar</packaging>
<name>kafka-protobuf-provider</name>

<dependencies>
  <dependency>
    <groupId>org.apache.kafka</groupId>
    <artifactId>kafka_${kafka.scala.version}</artifactId>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>com.squareup.wire</groupId>
    <artifactId>wire-schema</artifactId>
  </dependency>
  <dependency>
    <groupId>org.jetbrains.kotlin</groupId>
    <artifactId>kotlin-stdlib</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java-util</artifactId>
  </dependency>
  <dependency>

```

```

    <groupId>com.google.api.grpc</groupId>
    <artifactId>proto-google-common-protos</artifactId>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-client</artifactId>
    <version>${io.confluent.schema-registry.version}</version>
</dependency>
<dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-protobuf-types</artifactId>
</dependency>
<dependency>
    <groupId>org.mockito</groupId>
    <artifactId>mockito-core</artifactId>
    <scope>test</scope>
</dependency>
<dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <scope>test</scope>
</dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <groupId>com.github.os72</groupId>
      <artifactId>protoc-jar-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
</project>

```

Found in path(s):

```

* /opt/cola/permits/1340188583_1654735047.5458267/0/kafka-protobuf-provider-6-2-0-jar/META-
INF/maven/io.confluent/kafka-protobuf-provider/pom.xml

```

1.178 opentelemetry 1.6.0

1.178.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Licensed under the Apache License, Version 2.0 (the "License");
```

```
// you may not use this file except in compliance with the License.
```

```
// You may obtain a copy of the License at
```

```
// http://www.apache.org/licenses/LICENSE-2.0
// distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/collector/logs/v1/logs_service.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/collector/metrics/v1/metrics_service.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/metrics/v1/metrics.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/metrics/experimental/metrics_config_service.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/common/v1/common.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/trace/v1/trace.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/resource/v1/resource.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/collector/trace/v1/trace_service.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/logs/v1/logs.proto
* /opt/cola/permits/1310721288_1650882615.53/0/opentelemetry-proto-1-6-0-alpha-jar/opentelemetry/proto/trace/v1/trace_config.proto
```

1.179 kafka-protobuf-serializer 6.2.0

1.179.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<parent>
```

```
<groupId>io.confluent</groupId>
```

```
<artifactId>kafka-schema-registry-parent</artifactId>
```

```
<version>6.2.0</version>
```

```
</parent>
```

```
<licenses>
```

```
<license>
```

```
<name>Confluent Community License</name>
```

```
<url>http://www.confluent.io/confluent-community-license</url>
```

```
<distribution>repo</distribution>
```

```
</license>
```



```
<license>
  <name>Apache License 2.0</name>
  <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
  <distribution>repo</distribution>
</license>
</licenses>
```

```
<artifactId>kafka-protobuf-serializer</artifactId>
<packaging>jar</packaging>
<name>kafka-protobuf-serializer</name>
```

```
<dependencies>
  <dependency>
    <groupId>org.apache.kafka</groupId>
    <artifactId>kafka_${kafka.scala.version}</artifactId>
    <scope>provided</scope>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-protobuf-provider</artifactId>
    <version>${io.confluent.schema-registry.version}</version>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-protobuf-types</artifactId>
  </dependency>
  <dependency>
    <groupId>com.google.api.grpc</groupId>
    <artifactId>proto-google-common-protos</artifactId>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>com.google.protobuf</groupId>
    <artifactId>protobuf-java-util</artifactId>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-serializer</artifactId>
    <version>${io.confluent.schema-registry.version}</version>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-client</artifactId>
    <version>${io.confluent.schema-registry.version}</version>
  </dependency>
  <dependency>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry</artifactId>
```

```

    <version>${io.confluent.schema-registry.version}</version>
    <scope>test</scope>
</dependency>
<dependency>
  <groupId>io.confluent</groupId>
  <artifactId>kafka-schema-registry</artifactId>
  <version>${io.confluent.schema-registry.version}</version>
  <type>test-jar</type>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>connect-api</artifactId>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>kafka-clients</artifactId>
  <classifier>test</classifier>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.apache.kafka</groupId>
  <artifactId>kafka_${kafka.scala.version}</artifactId>
  <classifier>test</classifier>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-core</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <groupId>com.github.os72</groupId>
      <artifactId>protoc-jar-maven-plugin</artifactId>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>

```

```

    <version>2.6</version>
    <executions>
      <execution>
        <goals>
          <goal>test-jar</goal>
        </goals>
        <phase>test-compile</phase>
      </execution>
    </executions>
  </plugin>
</plugins>
</build>
</project>

```

Found in path(s):

* /opt/cola/permits/1340188649_1654735015.0795474/0/kafka-protobuf-serializer-6-2-0-jar/META-INF/maven/io.confluent/kafka-protobuf-serializer/pom.xml

1.180 netty-project 4.1.74.Final

1.180.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Implementation-Title: Netty/Codec

Bundle-Description: Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers and clients.

Automatic-Module-Name: io.netty.codec

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0>

Bundle-SymbolicName: io.netty.codec

Implementation-Version: 4.1.74.Final

Built-By: norman

Bnd-LastModified: 1644314286237

Bundle-ManifestVersion: 2

Implementation-Vendor-Id: io.netty

Bundle-DocURL: <https://netty.io/>

Bundle-Vendor: The Netty Project

Import-Package: com.aayushatharva.brotli4j;resolution:=optional,com.aayushatharva.brotli4j.decoder;resolution:=optional,com.aayushatharva.brotli4j.encoder;resolution:=optional,com.github.luben.zstd;version="[1.5,2)";resolution:=optional,com.google.protobuf;version="[2.6,3)";resolution:=optional,com.google.protobuf.nano;resolution:=optional,com.jcraft.jzlib;resolution:=optional,com.ning.compress;version="[1.0,2)";resolution:=optional,com.ning.compress.lzf;version="[1.0,2)";resolution:=optional,com.ning.compress.lzf.util;version="[1.0,2)";resolution:=optional,io.netty.buffer;version="[4.1,5)",io.netty.channel;version

="[4.1,5)",io.netty.channel.socket;version="[4.1,5)",io.netty.util;version="[4.1,5)",io.netty.util.concurrent;version="[4.1,5)",io.netty.util.internal;version="[4.1,5)",io.netty.util.internal.logging;version="[4.1,5)",lzma.sdk;resolution:=optional,lzma.sdk.lzma;resolution:=optional,net.jpountz.lz4;resolution:=optional,net.jpountz.xxhash;resolution:=optional,org.jboss.marshalling;resolution:=optional,sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="[1,2)";resolution:=optional

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.6))"

Tool: Bnd-2.4.1.201501161923

Implementation-Vendor: The Netty Project

Export-Package: io.netty.handler.codec;uses:="io.netty.buffer,io.netty.channel,io.netty.channel.socket,io.netty.util";version="4.1.74",io.netty.handler.codec.base64;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.bytes;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.compression;uses:="com.aayushatharva.brotli4j.encoder,io.netty.buffer,io.netty.channel,io.netty.handler.codec,net.jpountz.lz4";version="4.1.74",io.netty.handler.codec.json;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.marshalling;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec,org.jboss.marshalling";version="4.1.74",io.netty.handler.codec.protobuf;uses:="com.google.protobuf,com.google.protobuf.nano,io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.serialization;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.string;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74",io.netty.handler.codec.xml;uses:="io.netty.buffer,io.netty.channel,io.netty.handler.codec";version="4.1.74"

Bundle-Name: Netty/Codec

Bundle-Version: 4.1.74.Final

Created-By: Apache Maven Bundle Plugin

Build-Jdk: 1.8.0_252

Implementation-URL: <https://netty.io/netty-codec/>

Found in path(s):

* /opt/cola/permits/1288694964_1648177303.82/0/netty-codec-4-1-74-final-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

~ Copyright 2012 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE>

2.0

~

- ~ Unless required by applicable law or agreed to in writing, software
- ~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
- ~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
- ~ License for the specific language governing permissions and limitations
- ~ under the License.

Found in path(s):

* /opt/cola/permits/1288694964_1648177303.82/0/netty-codec-4-1-74-final-jar/META-INF/maven/io.netty/netty-codec/pom.xml

1.181 jetbrains-kotlin-kotlin-stdlib-jdk7 1.3.11

1.181.1 Available under license :

The version of Rhino used in GWT is licensed under a dual license,
Netscape Public License 1.1 / GNU General Public License.

The text of the Netscape Public License

is provided below (<http://website-archive.mozilla.org/www.mozilla.org/mpl/MPL/NPL/1.1/>):

AMENDMENTS

The Netscape Public License Version 1.1 ("NPL") consists of the Mozilla Public License Version 1.1 with the following Amendments, including Exhibit A-Netscape Public License. Files identified with "Exhibit A-Netscape Public License" are governed by the Netscape Public License Version 1.1.

Additional Terms applicable to the Netscape Public License.

I. Effect.

These additional terms described in this Netscape Public License -- Amendments shall apply to the Mozilla Communicator client code and to all Covered Code under this License.

II. "Netscape's Branded Code" means Covered Code that Netscape distributes and/or permits others to distribute under one or more trademark(s) which are controlled by Netscape but which are not licensed for use under this License.

III. Netscape and logo.

This License does not grant any rights to use the trademarks "Netscape", the "Netscape N and horizon" logo or the "Netscape lighthouse" logo, "Netcenter", "Gecko", "Java" or "JavaScript", "Smart Browsing" even if such marks are included

in the Original Code or Modifications.

IV. Inability to Comply Due to Contractual Obligation.

Prior to licensing the Original Code under this License, Netscape has licensed third party code for use in Netscape's Branded Code. To the extent that Netscape is limited contractually from making such third party code available under this License, Netscape may choose to reintegrate such code into Covered Code without being required to distribute such code in Source Code form, even if such code would otherwise be considered "Modifications" under this License.

V. Use of Modifications and Covered Code by Initial Developer.

V.1. In General.

The obligations of Section 3 apply to Netscape, except to the extent specified in this Amendment, Section V.2 and V.3.

V.2. Other Products.

Netscape may include Covered Code in products other than the Netscape's Branded Code which are released by Netscape during the two (2) years following the release date of the Original Code, without such additional products becoming subject to the terms of this License, and may license such additional products on different terms from those contained in this License.

V.3. Alternative Licensing.

Netscape may license the Source Code of Netscape's Branded Code, including Modifications incorporated therein, without such Netscape Branded Code becoming subject to the terms of this License, and may license such Netscape Branded Code on different terms from those contained in this License.

VI. Litigation.

Notwithstanding the limitations of Section 11 above, the provisions regarding litigation in Section 11(a), (b) and (c) of the License shall apply to all disputes relating to this License.

EXHIBIT A-Netscape Public License.

"The contents of this file are subject to the Netscape Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1998-1999 Netscape Communications Corporation. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the [____] License."

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or

hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is

granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the

party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN

ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections

2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do

business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as Multiple-Licensed. Multiple-Licensed means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.
_____. Portions created by _____
_____ are Copyright (C) _____

_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [_____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [_____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program

is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is

implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE

PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

```
=====  
/**  
 * Copyright 2010 Tim Down.  
 *  
 * Licensed under the Apache License, Version 2.0 (the "License");  
 * you may not use this file except in compliance with the License.  
 * You may obtain a copy of the License at  
 *  
 * http://www.apache.org/licenses/LICENSE-2.0  
 *  
 * Unless required by applicable law or agreed to in writing, software  
 * distributed under the License is distributed on an "AS IS" BASIS,  
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
 * See the License for the specific language governing permissions and  
 * limitations under the License.  
 */
```

```
=====  
== NOTICE file corresponding to the section 4 d of ==  
== the Apache License, Version 2.0, ==  
== in this case for the Kotlin Compiler distribution. ==  
=====
```

Kotlin Compiler

Copyright 2010-2017 JetBrains s.r.o and respective authors and developers
The Trove library is licensed under the Lesser GNU Public License,
which is included with the distribution in a file called trove_license.txt.

The PrimeFinder and HashFunctions classes in Trove are subject to the
following license restrictions:

Copyright (c) 1999 CERN - European Organization for Nuclear Research.

Permission to use, copy, modify, distribute and sell this software and
its documentation for any purpose is hereby granted without fee,
provided that the above copyright notice appear in all copies and that
both that copyright notice and this permission notice appear in
supporting documentation. CERN makes no representations about the
suitability of this software for any purpose. It is provided "as is"
without expressed or implied warranty.

Copyright (c) 2005-2010 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

/*

* Copyright 2010-2018 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright (C) 2009-2015 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

SCALA LICENSE

Copyright (c) 2002-2012 EPFL, Lausanne, unless otherwise specified.
All rights reserved.

This software was developed by the Programming Methods Laboratory of the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.

Permission to use, copy, modify, and distribute this software in source or binary form for any purpose with or without fee is hereby granted, provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the EPFL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or

promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Apache 2 license (given in full in LICENSE.txt) applies to all code in this repository which is copyright by JetBrains. The following sections of the repository contain third-party code, to which different licenses may apply:

Kotlin Compiler

The following modules contain third-party code and are incorporated into the Kotlin compiler and/or the Kotlin IntelliJ IDEA plugin:

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxStackSizeAndLocalsCalculator.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxLocalsCalculator.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/MethodAnalyzer.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: core/reflection/jvm/src/kotlin.reflect/jvm/internal/pcollections
 - License: MIT ([license/third_party/pcollections_LICENSE.txt][pcollections])
 - Origin: Derived from PCollections, A Persistent Java Collections Library (<https://pcollections.org/>)
- Path: eval4j/src/org/jetbrains/eval4j/interpreterLoop.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/OptimizationBasicInterpreter.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-

- Path: js/js.ast
 - License: BSD ([license/third_party/dart_LICENSE.txt][dart])
 - Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,

- Path: js/js.inliner/src/org/jetbrains/kotlin/js/inline/FunctionInlineMutator.kt
 - License: BSD ([license/third_party/dart_LICENSE.txt][dart])
 - Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,

- Path: js/js.parser/src/com/google
 - License: Netscape Public License 1.1 ([license/third_party/rhino_LICENSE.txt][rhino])
 - Origin: Originally part of GWT, (C) 2007-08 Google Inc., distributed under the Apache 2 license. The code is derived from Rhino, (C) 1997-1999 Netscape Communications Corporation, distributed under the Netscape Public License.

- Path: js/js.translator/qunit/qunit.js
 - License: MIT ([license/third_party/qunit_license.txt][qunit])
 - Origin: QUnit, Copyright (c) 2012 John Resig, Jrn Zaefferer,

- Path: libraries/stdlib/src/kotlin/collections
 - License: Apache 2 ([license/third_party/gwt_license.txt][gwt])
 - Origin: Derived from GWT, (C) 2007-08 Google Inc.

- Path: libraries/stdlib/unsigned/src/kotlin/UnsignedUtils.kt
 - License: Apache 2 ([license/third_party/guava_license.txt][guava])
 - Origin: Derived from Guava's UnsignedLongs, (C) 2011 The Guava Authors

- Path: libraries/stdlib/jvm/src/kotlin/util/MathJVM.kt
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.

- Path: libraries/stdlib/js/src/kotlin/collections
 - License: Apache 2 ([license/third_party/gwt_license.txt][gwt])
 - Origin: Derived from GWT, (C) 2007-08 Google Inc.

- Path: libraries/stdlib/js/src/js/long.js
 - License: Apache 2 ([license/third_party/closure-compiler_LICENSE.txt][closure-compiler])
 - Origin: Google Closure Library, Copyright 2009 The Closure Library Authors

- Path: libraries/stdlib/js/src/js/polyfills.js
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.

- Path: plugins/lint/android-annotations
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: plugins/lint/lint-api
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: plugins/lint/lint-checks
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: plugins/lint/lint-idea
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

Kotlin Test Data

The following source code is used for testing the Kotlin compiler and/or plugin and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: third-party/annotations/android
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: third-party/annotations/com/android
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: third-party/annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.
- Path: third-party/annotations/androidx
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project
- Path: third-party/annotations/edu/umd/cs/findbugs
 - License: LGPL 2.1 ([license/third_party/testdata/findbugs_license.txt][findbugs])
 - Origin: Bytecode Analysis Framework, Copyright (C) 2005 University of Maryland
- Path: third-party/jdk8-annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.
- Path: third-party/annotations/io/reactivex
 - License: Apache 2 ([license/third_party/testdata/rxjava_license.txt][rxjava])
 - Origin: RxJava, Copyright (c) 2016-present, RxJava Contributors
- Path: third-party/annotations/lombok
 - License: MIT ([license/third_party/testdata/lombok_license.txt][lombok])
 - Origin: Project Lombok, Copyright (C) 2009-2013 The Project Lombok Authors

- Path: idea/idea-android/tests/org/jetbrains/kotlin/android/AndroidTestBase.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lintQuickfix/requiresApi/RequiresApi.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/IntRange.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/RequiresPermission.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: libraries/tools/kotlin-gradle-plugin-integration-
tests/src/test/resources/testProject/allOpenSpring/src/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/AndroidDaggerProject
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/kapt2
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.

- Path: libraries/tools/kotlin-maven-plugin-test/src/it/test-allopen-
spring/src/main/java/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

Kotlin Tools and Libraries Tests

The following source code is used for testing the Kotlin tools and/or libraries and is not incorporated into any distributions of the tools or libraries:

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/kotlin/org/jetbrains/kotlin/gradle/PluginsDslIT.kt
 - License: Apache 2 ([license/third_party/testdata/gradle_license.txt][gradle])
 - Origin: Gradle, Copyright 2002-2017 Gradle, Inc.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/target/test-
classes/testProject/noArgJpa/src/javax/persistence/Entity.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
and Eclipse Distribution License - v1.0

([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])

- Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

- Path: libraries/tools/kotlin-gradle-plugin-integration-

tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Entity.java

- License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])

and Eclipse Distribution License - v1.0

([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])

- Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

- Path: libraries/tools/kotlin-gradle-plugin-integration-

tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Embeddable.java

- License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])

and Eclipse Distribution License - v1.0

([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])

- Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

Example Code

The following code is provided as examples and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: libraries/examples/browser-example/src/js/jquery.js

- License: MIT ([license/third_party/jquery_license.txt][jquery])

- Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

- Path: libraries/examples/browser-example-with-library/src/js/jquery.js

- License: MIT ([license/third_party/jquery_license.txt][jquery])

- Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

[aosp]: third_party/aosp_license.txt

[asm]: third_party/asm_license.txt

[boost]: third_party/boost_LICENSE.txt

[closure-compiler]: third_party/closure-compiler_LICENSE.txt

[dagger]: third_party/testdata/dagger_license.txt

[dart]: third_party/dart_LICENSE.txt

[eclipse]: third_party/testdata/eclipse_license.txt

[eclipse-distribution]: third_party/testdata/eclipse_distribution_license.txt

[findbugs]: third_party/testdata/findbugs_license.txt

[gradle]: third_party/testdata/gradle_license.txt

[guava]: third_party/guava_license.txt

[gwt]: third_party/gwt_license.txt

[jquery]: third_party/jquery_license.txt

[lombok]: third_party/testdata/lombok_license.txt

[pcollections]: third_party/pcollections_LICENSE.txt

[qunit]: third_party/qunit_license.txt

[rhino]: third_party/rhino_LICENSE.txt

[rxjava]: third_party/testdata/rxjava_license.txt

[spring]: third_party/testdata/spring_license.txt

This license applies to all parts of Dart that are not externally maintained libraries. The external maintained libraries used by Dart are:

7-Zip - in third_party/7zip
JSCRE - in runtime/third_party/jscre
Ant - in third_party/apache_ant
args4j - in third_party/args4j
bzip2 - in third_party/bzip2
dromaeo - in samples/third_party/dromaeo
Eclipse - in third_party/eclipse
gsutil = in third_party/goutil
Guava - in third_party/guava
hamcrest - in third_party/hamcrest
HttpLib2 - in samples/third_party/httpLib2
JSON - in third_party/json
JUnit - in third_party/junit
Oauth - in samples/third_party/oauth2client
Rhino - in third_party/rhino
weberknecht - in third_party/weberknecht

The libraries may have their own licenses; we recommend you read them, as their terms may differ from the terms below.

Copyright 2012, the Dart project authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE

OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BSD License

Copyright (c) 2000-2006, www.hamcrest.org

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====
Gradle Subcomponents:

License for the slf4j package

SLF4J License

Copyright (c) 2004-2007 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

These terms are identical to those of the MIT License, also called the X License or the X11 License, which is a simple, permissive non-copyleft free software license. It is deemed compatible with virtually all types of licenses, commercial or otherwise. In particular, the Free Software Foundation has declared it compatible with GNU GPL. It is also known to be approved by the Apache Software Foundation as compatible with Apache Software License.

License for the JUnit package

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to

Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the

originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the

intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

License for the JCIFS package

JCIFS License

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get
it if you want it; that you can change the software and use pieces of
it in new free programs; and that you are informed that you can do
these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it

does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the

Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility

is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work

under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is

normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further

restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the

ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

Google Dart Js backend was removed - <https://code.google.com/p/dart/source/detail?r=4771>

According to <http://www.apache.org/legal/3party.html> we can include "Google Dart Js backend" in source form, because code license is "New BSD License" (Authorized License).

This part of code will be removed when kotlin will be rewritten on kotlin.

Copyright JS Foundation and other contributors, <https://js.foundation/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/jquery>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER

DEALINGS IN THE SOFTWARE.

/*

* Copyright 2010-2018 JetBrains s.r.o. Use of this source code is governed by the Apache 2.0 license
* that can be found in the license/LICENSE.txt file.

*/

```
package org.jetbrains.idl2k.util
```

```
import org.xml.sax.InputSource
```

```
import java.io.File
```

```
import javax.xml.xpath.XPathFactory
```

```
fun readCopyrightNoticeFromProfile(copyrightProfile: File): String {
```

```
    val template = copyrightProfile.reader().use { reader ->
```

```
        XPathFactory.newInstance().newXPath().evaluate("/component/copyright/option[@name='notice']/@value",
```

```
InputSource(reader))
```

```
    }
```

```
    val yearTemplate = "&#36;today.year"
```

```
    val year = java.time.LocalDate.now().year.toString()
```

```
    assert(yearTemplate in template)
```

```
    return template.replace(yearTemplate, year).lines().joinToString("", prefix = "*/\n", postfix = " *\n") { " * $it\n" }
```

```
}
```

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using

a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License").

Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated

/*

* Copyright 2010-2017 JetBrains s.r.o.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact

all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest

your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License.

Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.

b) Give prominent notice with the combined library of the fact

that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the

integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE

IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
The files in this package are taken from mozilla's Rhino project.
See <http://www.mozilla.org/rhino/>

The files modified from Rhino 1.5R3
(<ftp://ftp.mozilla.org/pub/js/rhino15R3.zip>).

--

The contents of this package are subject to the Netscape Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Rhino code, released
May 6, 1999.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1997-2000 Netscape Communications Corporation. All Rights Reserved.

Alternatively, the contents of this file may be used under the terms of the GNU Public License (the "GPL"), in which case the provisions of the GPL are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the GPL and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the GPL. If you do not delete the provisions above, a recipient may use your version of this

file under either the NPL or the GPL.

Copyright (c) 2008 Harold Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Eclipse Public License, Version 1.0 (EPL-1.0)

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are

necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

- a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

- b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.
No hardware per se is licensed hereunder.

- c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

- d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
 - iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims,

lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

- a) promptly notify the Commercial Contributor in writing of such claim,
- and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

The MIT License

Copyright (c) 2003, Kohsuke Kawaguchi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright JS Foundation and other contributors, <https://js.foundation>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/qunitjs/qunit>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the

library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from

such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the

Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under

Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on

the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to

refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only

on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.182 dropwizard-request-logging-support 2.0.18

1.182.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.183 kotlin-stdlib 1.3.71

1.183.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2016 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Library.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/String.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/reflect/KClassifier.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/reflect/KVariance.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-
jar/kotlin/internal/InternalAnnotations.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/ArrayIntrinsics.kt
* /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/reflect/KVisibility.kt
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2010-2015 JetBrains s.r.o.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

*/

Found in path(s):

- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Char.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Number.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Iterator.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Nothing.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/CharSequence.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Comparable.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/reflect/KParameter.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Any.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Unit.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Throwable.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Function.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Boolean.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/Collections.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/annotation/Annotations.kt
- * /opt/cola/permits/1136846901_1613931357.97/0/kotlin-stdlib-1-3-71-sources-2-jar/kotlin/reflect/KDeclarationContainer.kt

1.184 apache-httpcomponents-core 4.4.9

1.184.1 Available under license :

Apache HttpCore

Copyright 2005-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.185 opentracing-util 0.33.0

1.185.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

Copyright 2016-2019 The OpenTracing Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188715_1654650550.7780979/0/opentracing-util-0-33-0-sources-2-jar/META-INF/maven/io.opentracing/opentracing-util/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016-2019 The OpenTracing Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188715_1654650550.7780979/0/opentracing-util-0-33-0-sources-2-jar/io/opentracing/util/ThreadLocalScope.java

* /opt/cola/permits/1340188715_1654650550.7780979/0/opentracing-util-0-33-0-sources-2-jar/io/opentracing/util/ThreadLocalScopeManager.java

* /opt/cola/permits/1340188715_1654650550.7780979/0/opentracing-util-0-33-0-sources-2-jar/io/opentracing/util/GlobalTracer.java

1.186 antlr 4.7.2

1.186.1 Available under license :

No license file was found, but licenses were detected in source scan.

~ Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/META-INF/maven/org.antlr/antlr4/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** The definitive ANTLR v3 tree grammar to walk/visit ANTLR v4 grammars.

* Parses trees created by ANTLRParser.g.

*

* Rather than have multiple tree grammars, one for each visit, I'm

```
* creating this generic visitor that knows about context. All of the
* boilerplate pattern recognition is done here. Then, subclasses can
* override the methods they care about. This prevents a lot of the same
* context tracking stuff like "set current alternative for current
* rule node" that is repeated in lots of tree filters.
```

```
*/
```

```
tree grammar GrammarTreeVisitor;
```

```
options {
```

```
language    = Java;
```

```
tokenVocab  = ANTLRParser;
```

```
ASTLabelType = GrammarAST;
```

```
}
```

```
// Include the copyright in this source and also the generated source
```

```
@header {
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2011 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
package org.antlr.v4.parse;
```

```
import org.antlr.v4.Tool;
```

```
import org.antlr.v4.tool.*;
```

```
import org.antlr.v4.tool.ast.*;
```

```
import java.lang.reflect.InvocationTargetException;
```

```

import java.lang.reflect.Method;
}

@members {
public String grammarName;
public GrammarAST currentRuleAST;
public String currentModeName = LexerGrammar.DEFAULT_MODE_NAME;
public String currentRuleName;
public GrammarAST currentOuterAltRoot;
public int currentOuterAltNumber = 1; // 1..n
public int rewriteEBNFLevel = 0;

public GrammarTreeVisitor() { this(null); }

// Should be abstract but can't make gen'd parser abstract;
// subclasses should implement else everything goes to stderr!
public ErrorManager getErrorManager() { return null; }

public void visitGrammar(GrammarAST t) { visit(t, "grammarSpec"); }
public void visit(GrammarAST t, String ruleName) {
    CommonTreeNodeStream nodes = new CommonTreeNodeStream(new GrammarASTAdaptor(), t);
    setTreeNodeStream(nodes);
    try {
        Method m = getClass().getMethod(ruleName);
        m.invoke(this);
    }
    catch (Throwable e) {
        ErrorManager errMgr = getErrorManager();
        if ( e instanceof InvocationTargetException ) {
            e = e.getCause();
        }
        //e.printStackTrace(System.err);
        if ( errMgr==null ) {
            System.err.println("can't find rule "+ruleName+
                " or tree structure error: "+t.toStringTree()
            );
            e.printStackTrace(System.err);
        }
        else errMgr.toolError(ErrorType.INTERNAL_ERROR, e);
    }
}

public void discoverGrammar(GrammarRootAST root, GrammarAST ID) { }
public void finishPrequels(GrammarAST firstPrequel) { }
public void finishGrammar(GrammarRootAST root, GrammarAST ID) { }

public void grammarOption(GrammarAST ID, GrammarAST valueAST) { }
public void ruleOption(GrammarAST ID, GrammarAST valueAST) { }

```

```

public void blockOption(GrammarAST ID, GrammarAST valueAST) { }
public void defineToken(GrammarAST ID) { }
public void defineChannel(GrammarAST ID) { }
public void globalNamedAction(GrammarAST scope, GrammarAST ID, ActionAST action) { }
public void importGrammar(GrammarAST label, GrammarAST ID) { }

public void modeDef(GrammarAST m, GrammarAST ID) { }

public void discoverRules(GrammarAST rules) { }
public void finishRules(GrammarAST rule) { }
public void discoverRule(RuleAST rule, GrammarAST ID, List<GrammarAST> modifiers,
    ActionAST arg, ActionAST returns, GrammarAST thrws,
    GrammarAST options, ActionAST locals,
    List<GrammarAST> actions,
    GrammarAST block) { }
public void finishRule(RuleAST rule, GrammarAST ID, GrammarAST block) { }
public void discoverLexerRule(RuleAST rule, GrammarAST ID, List<GrammarAST> modifiers,
    GrammarAST block) { }
public void finishLexerRule(RuleAST rule, GrammarAST ID, GrammarAST block) { }
public void ruleCatch(GrammarAST arg, ActionAST action) { }
public void finallyAction(ActionAST action) { }
public void discoverOuterAlt(AltAST alt) { }
public void finishOuterAlt(AltAST alt) { }
public void discoverAlt(AltAST alt) { }
public void finishAlt(AltAST alt) { }

public void ruleRef(GrammarAST ref, ActionAST arg) { }
public void tokenRef(TerminalAST ref) { }
public void elementOption(GrammarASTWithOptions t, GrammarAST ID, GrammarAST valueAST) { }
public void stringRef(TerminalAST ref) { }
public void wildcardRef(GrammarAST ref) { }
public void actionInAlt(ActionAST action) { }
public void sempredInAlt(PredAST pred) { }
public void label(GrammarAST op, GrammarAST ID, GrammarAST element) { }
public void lexerCallCommand(int outerAltNumber, GrammarAST ID, GrammarAST arg) { }
public void lexerCommand(int outerAltNumber, GrammarAST ID) { }

protected void enterGrammarSpec(GrammarAST tree) { }
protected void exitGrammarSpec(GrammarAST tree) { }

protected void enterPrequelConstructs(GrammarAST tree) { }
protected void exitPrequelConstructs(GrammarAST tree) { }

protected void enterPrequelConstruct(GrammarAST tree) { }
protected void exitPrequelConstruct(GrammarAST tree) { }

protected void enterOptionsSpec(GrammarAST tree) { }
protected void exitOptionsSpec(GrammarAST tree) { }

```



```
protected void enterOption(GrammarAST tree) { }
protected void exitOption(GrammarAST tree) { }

protected void enterOptionValue(GrammarAST tree) { }
protected void exitOptionValue(GrammarAST tree) { }

protected void enterDelegateGrammars(GrammarAST tree) { }
protected void exitDelegateGrammars(GrammarAST tree) { }

protected void enterDelegateGrammar(GrammarAST tree) { }
protected void exitDelegateGrammar(GrammarAST tree) { }

protected void enterTokensSpec(GrammarAST tree) { }
protected void exitTokensSpec(GrammarAST tree) { }

protected void enterTokenSpec(GrammarAST tree) { }
protected void exitTokenSpec(GrammarAST tree) { }

protected void enterChannelsSpec(GrammarAST tree) { }
protected void exitChannelsSpec(GrammarAST tree) { }

protected void enterChannelSpec(GrammarAST tree) { }
protected void exitChannelSpec(GrammarAST tree) { }

protected void enterAction(GrammarAST tree) { }
protected void exitAction(GrammarAST tree) { }

protected void enterRules(GrammarAST tree) { }
protected void exitRules(GrammarAST tree) { }

protected void enterMode(GrammarAST tree) { }
protected void exitMode(GrammarAST tree) { }

protected void enterLexerRule(GrammarAST tree) { }
protected void exitLexerRule(GrammarAST tree) { }

protected void enterRule(GrammarAST tree) { }
protected void exitRule(GrammarAST tree) { }

protected void enterExceptionGroup(GrammarAST tree) { }
protected void exitExceptionGroup(GrammarAST tree) { }

protected void enterExceptionHandler(GrammarAST tree) { }
protected void exitExceptionHandler(GrammarAST tree) { }

protected void enterFinallyClause(GrammarAST tree) { }
protected void exitFinallyClause(GrammarAST tree) { }
```

```
protected void enterLocals(GrammarAST tree) { }
protected void exitLocals(GrammarAST tree) { }

protected void enterRuleReturns(GrammarAST tree) { }
protected void exitRuleReturns(GrammarAST tree) { }

protected void enterThrowsSpec(GrammarAST tree) { }
protected void exitThrowsSpec(GrammarAST tree) { }

protected void enterRuleAction(GrammarAST tree) { }
protected void exitRuleAction(GrammarAST tree) { }

protected void enterRuleModifier(GrammarAST tree) { }
protected void exitRuleModifier(GrammarAST tree) { }

protected void enterLexerRuleBlock(GrammarAST tree) { }
protected void exitLexerRuleBlock(GrammarAST tree) { }

protected void enterRuleBlock(GrammarAST tree) { }
protected void exitRuleBlock(GrammarAST tree) { }

protected void enterLexerOuterAlternative(AltAST tree) { }
protected void exitLexerOuterAlternative(AltAST tree) { }

protected void enterOuterAlternative(AltAST tree) { }
protected void exitOuterAlternative(AltAST tree) { }

protected void enterLexerAlternative(GrammarAST tree) { }
protected void exitLexerAlternative(GrammarAST tree) { }

protected void enterLexerElements(GrammarAST tree) { }
protected void exitLexerElements(GrammarAST tree) { }

protected void enterLexerElement(GrammarAST tree) { }
protected void exitLexerElement(GrammarAST tree) { }

protected void enterLabeledLexerElement(GrammarAST tree) { }
protected void exitLabeledLexerElement(GrammarAST tree) { }

protected void enterLexerBlock(GrammarAST tree) { }
protected void exitLexerBlock(GrammarAST tree) { }

protected void enterLexerAtom(GrammarAST tree) { }
protected void exitLexerAtom(GrammarAST tree) { }

protected void enterActionElement(GrammarAST tree) { }
protected void exitActionElement(GrammarAST tree) { }
```

```
protected void enterAlternative(AltAST tree) { }
protected void exitAlternative(AltAST tree) { }

protected void enterLexerCommand(GrammarAST tree) { }
protected void exitLexerCommand(GrammarAST tree) { }

protected void enterLexerCommandExpr(GrammarAST tree) { }
protected void exitLexerCommandExpr(GrammarAST tree) { }

protected void enterElement(GrammarAST tree) { }
protected void exitElement(GrammarAST tree) { }

protected void enterAstOperand(GrammarAST tree) { }
protected void exitAstOperand(GrammarAST tree) { }

protected void enterLabeledElement(GrammarAST tree) { }
protected void exitLabeledElement(GrammarAST tree) { }

protected void enterSubrule(GrammarAST tree) { }
protected void exitSubrule(GrammarAST tree) { }

protected void enterLexerSubrule(GrammarAST tree) { }
protected void exitLexerSubrule(GrammarAST tree) { }

protected void enterBlockSuffix(GrammarAST tree) { }
protected void exitBlockSuffix(GrammarAST tree) { }

protected void enterEbnfSuffix(GrammarAST tree) { }
protected void exitEbnfSuffix(GrammarAST tree) { }

protected void enterAtom(GrammarAST tree) { }
protected void exitAtom(GrammarAST tree) { }

protected void enterBlockSet(GrammarAST tree) { }
protected void exitBlockSet(GrammarAST tree) { }

protected void enterSetElement(GrammarAST tree) { }
protected void exitSetElement(GrammarAST tree) { }

protected void enterBlock(GrammarAST tree) { }
protected void exitBlock(GrammarAST tree) { }

protected void enterRuleref(GrammarAST tree) { }
protected void exitRuleref(GrammarAST tree) { }

protected void enterRange(GrammarAST tree) { }
protected void exitRange(GrammarAST tree) { }
```

```

protected void enterTerminal(GrammarAST tree) { }
protected void exitTerminal(GrammarAST tree) { }

protected void enterElementOptions(GrammarAST tree) { }
protected void exitElementOptions(GrammarAST tree) { }

protected void enterElementOption(GrammarAST tree) { }
protected void exitElementOption(GrammarAST tree) { }

@Override
public void traceIn(String ruleName, int ruleIndex) {
    System.err.println("enter "+ruleName+": "+input.LT(1));
}

@Override
public void traceOut(String ruleName, int ruleIndex) {
    System.err.println("exit "+ruleName+": "+input.LT(1));
}
}

grammarSpec
@init {
    enterGrammarSpec($start);
}
@after {
    exitGrammarSpec($start);
}
: ^( GRAMMAR ID {grammarName=$ID.text;}
    {discoverGrammar((GrammarRootAST)$GRAMMAR, $ID);}
    prequelConstructs
    {finishPrequels($prequelConstructs.firstOne);}
    rules mode*
    {finishGrammar((GrammarRootAST)$GRAMMAR, $ID);}
)
;

prequelConstructs returns [GrammarAST firstOne=null]
@init {
    enterPrequelConstructs($start);
}
@after {
    exitPrequelConstructs($start);
}
: {$firstOne=$start;} prequelConstruct+
|
;

```

```

prequelConstruct
@init {
  enterPrequelConstructs($start);
}
@after {
  exitPrequelConstructs($start);
}
: optionsSpec
  | delegateGrammars
  | tokensSpec
  | channelsSpec
  | action
;

```

```

optionsSpec
@init {
  enterOptionsSpec($start);
}
@after {
  exitOptionsSpec($start);
}
: ^(OPTIONS option*)
;

```

```

option
@init {
  enterOption($start);
  boolean rule = inContext("RULE ...");
  boolean block = inContext("BLOCK ...");
}
@after {
  exitOption($start);
}
: ^(a=ASSIGN ID v=optionValue)
  {
    if ( block ) blockOption($ID, $v.start); // most specific first
    else if ( rule ) ruleOption($ID, $v.start);
    else grammarOption($ID, $v.start);
  }
;

```

```

optionValue returns [String v]
@init {
  enterOptionValue($start);
  $v = $start.token.getText();
}
@after {
  exitOptionValue($start);
}

```

```

}
: ID
| STRING_LITERAL
| INT
;

delegateGrammars
@init {
enterDelegateGrammars($start);
}
@after {
exitDelegateGrammars($start);
}
: ^(IMPORT delegateGrammar+)
;

```

```

delegateGrammar
@init {
enterDelegateGrammar($start);
}
@after {
exitDelegateGrammar($start);
}
: ^(ASSIGN label=ID id=ID) {importGrammar($label, $id);}
| id=ID {importGrammar(null, $id);}
;

```

```

tokensSpec
@init {
enterTokensSpec($start);
}
@after {
exitTokensSpec($start);
}
: ^(TOKENS_SPEC tokenSpec+)
;

```

```

tokenSpec
@init {
enterTokenSpec($start);
}
@after {
exitTokenSpec($start);
}
: ID {defineToken($ID);}
;

```

```

channelsSpec

```

```

@init {
  enterChannelsSpec($start);
}
@after {
  exitChannelsSpec($start);
}
: ^(CHANNELS channelSpec+)
;

channelSpec
@init {
  enterChannelSpec($start);
}
@after {
  exitChannelSpec($start);
}
: ID {defineChannel($ID);}
;

action
@init {
  enterAction($start);
}
@after {
  exitAction($start);
}
: ^(AT sc=ID? name=ID ACTION) {globalNamedAction($sc, $name, (ActionAST)$ACTION);}
;

rules
@init {
  enterRules($start);
}
@after {
  exitRules($start);
}
: ^(RULES {discoverRules($RULES);} (rule|lexerRule)* {finishRules($RULES);})
;

mode
@init {
  enterMode($start);
}
@after {
  exitMode($start);
}
: ^( MODE ID {currentModeName=$ID.text; modeDef($MODE, $ID);} lexerRule* )
;

```

```

lexerRule
@init {
    enterLexerRule($start);
    List<GrammarAST> mods = new ArrayList<GrammarAST>();
    currentOuterAltNumber=0;
}
@after {
    exitLexerRule($start);
}
: ^( RULE TOKEN_REF
    {currentRuleName=$TOKEN_REF.text; currentRuleAST=$RULE;}
    (^(RULEMODIFIERS m=FRAGMENT {mods.add($m);}))?
    {discoverLexerRule((RuleAST)$RULE, $TOKEN_REF, mods, (GrammarAST)input.LT(1));}
    lexerRuleBlock
    {
        finishLexerRule((RuleAST)$RULE, $TOKEN_REF, $lexerRuleBlock.start);
        currentRuleName=null; currentRuleAST=null;
    }
    )
;

rule
@init {
    enterRule($start);
    List<GrammarAST> mods = new ArrayList<GrammarAST>();
    List<GrammarAST> actions = new ArrayList<GrammarAST>(); // track roots
    currentOuterAltNumber=0;
}
@after {
    exitRule($start);
}
: ^( RULE RULE_REF {currentRuleName=$RULE_REF.text; currentRuleAST=$RULE;}
    (^(RULEMODIFIERS (m=ruleModifier {mods.add($m.start);}+)))?
    ARG_ACTION?
    ret=ruleReturns?
    thr=throwsSpec?
    loc=locals?
    ( opts=optionsSpec
    | a=ruleAction {actions.add($a.start);}
    )*
    {discoverRule((RuleAST)$RULE, $RULE_REF, mods, (ActionAST)$ARG_ACTION,
        $ret.start!=null?(ActionAST)$ret.start.getChild(0):null,
        $thr.start, $opts.start,
        $loc.start!=null?(ActionAST)$loc.start.getChild(0):null,
        actions, (GrammarAST)input.LT(1));}
    ruleBlock exceptionGroup
    {finishRule((RuleAST)$RULE, $RULE_REF, $ruleBlock.start); currentRuleName=null; currentRuleAST=null;}

```



```

    )
    ;

exceptionGroup
@init {
    enterExceptionGroup($start);
}
@after {
    exitExceptionGroup($start);
}
: exceptionHandler* finallyClause?
;

exceptionHandler
@init {
    enterExceptionHandler($start);
}
@after {
    exitExceptionHandler($start);
}
: ^(CATCH ARG_ACTION ACTION) {ruleCatch($ARG_ACTION, (ActionAST)$ACTION);}
;

finallyClause
@init {
    enterFinallyClause($start);
}
@after {
    exitFinallyClause($start);
}
: ^(FINALLY ACTION) {finallyAction((ActionAST)$ACTION);}
;

locals
@init {
    enterLocals($start);
}
@after {
    exitLocals($start);
}
: ^(LOCALS ARG_ACTION)
;

ruleReturns
@init {
    enterRuleReturns($start);
}
@after {

```

```
exitRuleReturns($start);
}
: ^(RETURNS ARG_ACTION)
;
```

```
throwsSpec
@init {
  enterThrowsSpec($start);
}
@after {
  exitThrowsSpec($start);
}
: ^(THROWS ID+)
;
```

```
ruleAction
@init {
  enterRuleAction($start);
}
@after {
  exitRuleAction($start);
}
: ^(AT ID ACTION)
;
```

```
ruleModifier
@init {
  enterRuleModifier($start);
}
@after {
  exitRuleModifier($start);
}
: PUBLIC
| PRIVATE
| PROTECTED
| FRAGMENT
;
```

```
lexerRuleBlock
@init {
  enterLexerRuleBlock($start);
}
@after {
  exitLexerRuleBlock($start);
}
: ^( BLOCK
( {
  currentOuterAltRoot = (GrammarAST)input.LT(1);
```

```
currentOuterAltNumber++;
}
lexerOuterAlternative
)+
)
;
```

ruleBlock

```
@init {
enterRuleBlock($start);
}
@after {
exitRuleBlock($start);
}
: ^( BLOCK
( {
currentOuterAltRoot = (GrammarAST)input.LT(1);
currentOuterAltNumber++;
}
outerAlternative
)+
)
;
```

lexerOuterAlternative

```
@init {
enterLexerOuterAlternative((AltAST)$start);
discoverOuterAlt((AltAST)$start);
}
@after {
finishOuterAlt((AltAST)$start);
exitLexerOuterAlternative((AltAST)$start);
}
: lexerAlternative
;
```

outerAlternative

```
@init {
enterOuterAlternative((AltAST)$start);
discoverOuterAlt((AltAST)$start);
}
@after {
finishOuterAlt((AltAST)$start);
exitOuterAlternative((AltAST)$start);
}
: alternative
;
```

```
lexerAlternative
@init {
  enterLexerAlternative($start);
}
@after {
  exitLexerAlternative($start);
}
: ^(LEXER_ALT_ACTION lexerElements lexerCommand+)
  | lexerElements
;
```

```
lexerElements
@init {
  enterLexerElements($start);
}
@after {
  exitLexerElements($start);
}
: ^(ALT lexerElement+)
;
```

```
lexerElement
@init {
  enterLexerElement($start);
}
@after {
  exitLexerElement($start);
}
: labeledLexerElement
  | lexerAtom
  | lexerSubrule
  | ACTION    {actionInAlt((ActionAST)$ACTION);}
  | SEMPRED   {sempredInAlt((PredAST)$SEMPRED);}
  | ^(ACTION elementOptions) {actionInAlt((ActionAST)$ACTION);}
  | ^(SEMPRED elementOptions) {sempredInAlt((PredAST)$SEMPRED);}
  | EPSILON
;
```

```
labeledLexerElement
@init {
  enterLabeledLexerElement($start);
}
@after {
  exitLabeledLexerElement($start);
}
: ^((ASSIGN|PLUS_ASSIGN) ID (lexerAtom|block))
;
```

```
lexerBlock
@init {
  enterLexerBlock($start);
}
@after {
  exitLexerBlock($start);
}
: ^(BLOCK optionsSpec? lexerAlternative+)
;
```

```
lexerAtom
@init {
  enterLexerAtom($start);
}
@after {
  exitLexerAtom($start);
}
: terminal
| ^(NOT blockSet)
| blockSet
| ^(WILDCARD elementOptions)
| WILDCARD
| LEXER_CHAR_SET
| range
| ruleref
;
```

```
actionElement
@init {
  enterActionElement($start);
}
@after {
  exitActionElement($start);
}
: ACTION
| ^(ACTION elementOptions)
| SEMPRED
| ^(SEMPRED elementOptions)
;
```

```
alternative
@init {
  enterAlternative((AltAST)$start);
  discoverAlt((AltAST)$start);
}
@after {
  finishAlt((AltAST)$start);
}
```

```

exitAlternative((AltAST)$start);
}
: ^(ALT elementOptions? element+)
| ^(ALT elementOptions? EPSILON)
;

lexerCommand
@init {
enterLexerCommand($start);
}
@after {
exitLexerCommand($start);
}
: ^(LEXER_ACTION_CALL ID lexerCommandExpr)
  {lexerCallCommand(currentOuterAltNumber, $ID, $lexerCommandExpr.start);}
| ID
  {lexerCommand(currentOuterAltNumber, $ID);}
;

lexerCommandExpr
@init {
enterLexerCommandExpr($start);
}
@after {
exitLexerCommandExpr($start);
}
: ID
| INT
;

element
@init {
enterElement($start);
}
@after {
exitElement($start);
}
: labeledElement
| atom
| subrule
| ACTION    {actionInAlt((ActionAST)$ACTION);}
| SEMPRED   {sempredInAlt((PredAST)$SEMPRED);}
| ^(ACTION elementOptions) {actionInAlt((ActionAST)$ACTION);}
| ^(SEMPRED elementOptions) {sempredInAlt((PredAST)$SEMPRED);}
| range
| ^(NOT blockSet)
| ^(NOT block)
;

```

astOperand

```
@init {
  enterAstOperand($start);
}
@after {
  exitAstOperand($start);
}
: atom
| ^(NOT blockSet)
| ^(NOT block)
;
```

labeledElement

```
@init {
  enterLabeledElement($start);
}
@after {
  exitLabeledElement($start);
}
: ^((ASSIGN|PLUS_ASSIGN) ID element) {label($start, $ID, $element.start);}
;
```

subrule

```
@init {
  enterSubrule($start);
}
@after {
  exitSubrule($start);
}
: ^(blockSuffix block)
| block
;
```

lexerSubrule

```
@init {
  enterLexerSubrule($start);
}
@after {
  exitLexerSubrule($start);
}
: ^(blockSuffix lexerBlock)
| lexerBlock
;
```

blockSuffix

```
@init {
  enterBlockSuffix($start);
```

```
}
@after {
  exitBlockSuffix($start);
}
: ebnfSuffix
;
```

```
ebnfSuffix
@init {
  enterEbnfSuffix($start);
}
@after {
  exitEbnfSuffix($start);
}
: OPTIONAL
| CLOSURE
| POSITIVE_CLOSURE
;
```

```
atom
@init {
  enterAtom($start);
}
@after {
  exitAtom($start);
}
: ^(DOT ID terminal)
| ^(DOT ID ruleref)
| ^(WILDCARD elementOptions) { wildcardRef($WILDCARD);}
| WILDCARD {wildcardRef($WILDCARD);}
| terminal
| blockSet
| ruleref
;
```

```
blockSet
@init {
  enterBlockSet($start);
}
@after {
  exitBlockSet($start);
}
: ^(SET setElement+)
;
```

```
setElement
@init {
  enterSetElement($start);
}
```



```

}
@after {
  exitSetElement($start);
}
: ^(STRING_LITERAL elementOptions) {stringRef((TerminalAST)$STRING_LITERAL);}
| ^(TOKEN_REF elementOptions) {tokenRef((TerminalAST)$TOKEN_REF);}
| STRING_LITERAL {stringRef((TerminalAST)$STRING_LITERAL);}
| TOKEN_REF {tokenRef((TerminalAST)$TOKEN_REF);}
| ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
{
  stringRef((TerminalAST)$a);
  stringRef((TerminalAST)$b);
}
| LEXER_CHAR_SET
;

```

```

block
@init {
  enterBlock($start);
}
@after {
  exitBlock($start);
}
: ^(BLOCK optionsSpec? ruleAction* ACTION? alternative+)
;

```

```

ruleref
@init {
  enterRuleref($start);
}
@after {
  exitRuleref($start);
}
: ^(RULE_REF arg=ARG_ACTION? elementOptions?)
{
  ruleRef($RULE_REF, (ActionAST)$ARG_ACTION);
  if ( $arg!=null ) actionInAlt((ActionAST)$arg);
}
;

```

```

range
@init {
  enterRange($start);
}
@after {
  exitRange($start);
}
: ^(RANGE STRING_LITERAL STRING_LITERAL)

```

```

;

terminal
@init {
  enterTerminal($start);
}
@after {
  exitTerminal($start);
}
: ^(STRING_LITERAL elementOptions)
  {stringRef((TerminalAST)$STRING_LITERAL);}
| STRING_LITERAL {stringRef((TerminalAST)$STRING_LITERAL);}
| ^(TOKEN_REF elementOptions) {tokenRef((TerminalAST)$TOKEN_REF);}
| TOKEN_REF {tokenRef((TerminalAST)$TOKEN_REF);}
;

elementOptions
@init {
  enterElementOptions($start);
}
@after {
  exitElementOptions($start);
}
: ^(ELEMENT_OPTIONS elementOption[(GrammarASTWithOptions)$start.getParent()]*);
;

elementOption[GrammarASTWithOptions t]
@init {
  enterElementOption($start);
}
@after {
  exitElementOption($start);
}
: ID {elementOption(t, $ID, null);}
| ^(ASSIGN id=ID v=ID) {elementOption(t, $id, $v);}
| ^(ASSIGN ID v=STRING_LITERAL) {elementOption(t, $ID, $v);}
| ^(ASSIGN ID v=ACTION) {elementOption(t, $ID, $v);}
| ^(ASSIGN ID v=INT) {elementOption(t, $ID, $v);}
;

```

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/GrammarTreeVisitor.g

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

- * Copyright (c) 2012-2016 Sam Harwell
- * Copyright (c) 2015 Janyou
- * All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- *
- * 1. Redistributions of source code must retain the above copyright
- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in the
- * documentation and/or other materials provided with the distribution.
- * 3. The name of the author may not be used to endorse or promote products
- * derived from this software without specific prior written permission.
- *
- * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
- * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
- * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
- * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
- * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
- * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
- * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
- * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
- * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
- * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- */

```
SwiftTypeInitMap ::= [
  "Int": "0",
  "Int64": "0",
  "Float": "0.0",
  "Double": "0.0",
  "Bool": "false",
  default: "nil" // anything other than a primitive type is an object
]
SwiftTypeMap ::= [
  "int": "Int",
  "float": "Float",
  "long": "Int64",
  "double": "Double",
  "bool": "Bool",
  "boolean": "Bool",
  default : key
]
// args must be <object-model-object>, <fields-resulting-in-STs>

accessLevelOpenOK(obj) ::= "<obj.accessLevel; null=\"open\">"
```

```
accessLevelNotOpen(obj) ::= "<obj.accessLevel; null=\"public\">"
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
<if(file.genPackage)>  
<!package <file.genPackage>;!>  
<endif>  
<namedActions.header>  
import Antlr4  
  
<parser>  
>>
```

```
ListenerFile(file, header, namedActions) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
<if(file.genPackage)>  
<!package <file.genPackage>;!>  
<endif>  
<header>  
import Antlr4  
  
/**  
 * This interface defines a complete listener for a parse tree produced by  
 * { @link <file.parserName> }.  
 */  
<accessLevelNotOpen(file)> protocol <file.grammarName>Listener: ParseTreeListener {  
  <file.listenerNames: {Iname |  
  /**  
  <if(file.listenerLabelRuleNames.(Iname))>  
  * Enter a parse tree produced by the { @code <Iname>\}  
  * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>\}.  
  <else>  
  * Enter a parse tree produced by { @link <file.parserName>#<Iname>\}.  
  <endif>  
  - Parameters:  
  - ctx: the parse tree  
  */  
  func enter<Iname; format="cap">(_ ctx: <file.parserName>.<Iname; format="cap">Context)  
  /**  
  <if(file.listenerLabelRuleNames.(Iname))>  
  * Exit a parse tree produced by the { @code <Iname>\}  
  * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>\}.  
  <else>  
  * Exit a parse tree produced by { @link <file.parserName>#<Iname>\}.  
  <endif>  
  - Parameters:  
  - ctx: the parse tree  
  */
```

```

func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname; format="cap">Context)); separator="\n">
}
>>

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>

import Antlr4

/**
 * This class provides an empty implementation of { @link <file.grammarName>Listener },
 * which can be extended to create a listener which only needs to handle a subset
 * of the available methods.
 */
<accessLevelOpenOK(file)> class <file.grammarName>BaseListener: <file.grammarName>Listener {
    <accessLevelNotOpen(file)> init() { \ }
    <file.listenerNames:{lname |

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func enter<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \ }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func exit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) { \ } ; separator="\n">

/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func enterEveryRule(_ ctx: ParserRuleContext) { \ }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>

```

```

*/
<accessLevelOpenOK(file)> func exitEveryRule(_ ctx: ParserRuleContext) { }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func visitTerminal(_ node: TerminalNode) { }
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation does nothing.\</p>
 */
<accessLevelOpenOK(file)> func visitErrorNode(_ node: ErrorNode) { }
}
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
 * This interface defines a complete generic visitor for a parse tree produced
 * by { @link <file.parserName>}.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void} for
 * operations with no return type.
 */
<accessLevelOpenOK(file)> class <file.grammarName>Visitor\<T>: ParseTreeVisitor\<T> {
  <file.visitorNames:{ lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
 * Visit a parse tree produced by the { @code <lname>}
 * labeled alternative in { @link <file.parserName>#<file.visitorLabelRuleNames.(lname)>}}.
<else>
 * Visit a parse tree produced by { @link <file.parserName>#<lname>}}.
<endif>
- Parameters:
- ctx: the parse tree
- returns: the visitor result
*/
<accessLevelOpenOK(file)> func visit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) -> T {
  fatalError(#function + " must be overridden")

```

```

\}
}; separator="\n">
}
>>

```

```

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
<!package <file.genPackage>;!>
<endif>
<header>
import Antlr4

/**
 * This class provides an empty implementation of { @link <file.grammarName>Visitor},
 * which can be extended to create a visitor which only needs to handle a subset
 * of the available methods.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void} for
 * operations with no return type.
 */
<accessLevelOpenOK(file)> class <file.grammarName>BaseVisitor\<T>: AbstractParseTreeVisitor\<T> {
  <file.visitorNames:{lname |
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation returns the result of calling
 * { @link #visitChildren\} on { @code ctx\}.\</p>
 */
<accessLevelOpenOK(file)> func visit<lname; format="cap">(_ ctx: <file.parserName>.<lname;
format="cap">Context) -> T? { return visitChildren(ctx) \}; separator="\n">
}
>>

```

```

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

```

```

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

```

```

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
</*!/@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})!>
<accessLevelOpenOK(parser)> class <parser.name>: <superClass; null="Parser"> {

```

```

  internal static var _decisionToDFA: [DFA] = {
    var decisionToDFA = [DFA]()

```

```

    let length = <parser.name>._ATN.getNumberOfDecisions()
    for i in 0..<length {
        <!-- decisionToDFA[i] = DFA(<parser.name>._ATN.getDecisionState(i)!, i);!>
        decisionToDFA.append(DFA(<parser.name>._ATN.getDecisionState(i)!, i))
    }
    return decisionToDFA
}()

internal static let _sharedContextCache = PredictionContextCache()

<if(parser.tokens)>
<accessLevelNotOpen(parser)>
enum Tokens: Int {
    case EOF = -1, <parser.tokens:{k | <k> = <parser.tokens.(k)>}; separator=", ", wrap, anchor>
}
<endif>

<accessLevelNotOpen(parser)>
static let <parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>

<accessLevelNotOpen(parser)>
static let ruleNames: [String] = [
    <parser.ruleNames:{r | "<r">}; separator=", ", wrap, anchor>
]

<vocabulary(parser.literalNames, parser.symbolicNames,
    accessLevelNotOpen(parser))>

override <accessLevelOpenOK(parser)>
func getGrammarFileName() -> String { return "<parser.grammarFileName; format="java-escape">" }

override <accessLevelOpenOK(parser)>
func getRuleNames() -> [String] { return <parser.name>.ruleNames }

override <accessLevelOpenOK(parser)>
func getSerializedATN() -> String { return <parser.name>._serializedATN }

override <accessLevelOpenOK(parser)>
func getATN() -> ATN { return <parser.name>._ATN }

<namedActions.members>
<parser:(ctor)()>
<funcs; separator="\n">

<if(sempredFuncs)>
override <accessLevelOpenOK(parser)>
func sempred(_ _localctx:RuleContext?, _ ruleIndex: Int, _ predIndex: Int)throws -> Bool {
    switch (ruleIndex) {

```



```

    <parser.sempredFuncs.values:{f}
case <f.ruleIndex>:
return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)); separator="\n">
    default: return true
}
<!return true;!>
}
<sempredFuncs.values; separator="\n">
<endif>

```

```
<atn>
```

```

<accessLevelNotOpen(parser)>
static let _serializedATN = <parser.name>ATN().jsonString

```

```

<accessLevelNotOpen(parser)>
static let _ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
}
>>

```

```

vocabulary(literalNames, symbolicNames, accessLevel) ::= <<
private static let _LITERAL_NAMES: [String?] = [
    <literalNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
]
private static let _SYMBOLIC_NAMES: [String?] = [
    <symbolicNames:{t | <t>}; null="nil", separator=", ", wrap, anchor>
]
<accessLevel>
static let VOCABULARY = Vocabulary(_LITERAL_NAMES, _SYMBOLIC_NAMES)
>>

```

```

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
override <accessLevelOpenOK(parser)>
func action(_ _localctx: RuleContext?, _ ruleIndex: Int, _ actionIndex: Int) throws {
    switch (ruleIndex) {
    <recog.actionFuncs.values:{f}
case <f.ruleIndex>:
    <f.name>_action((_localctx as <f.ctxType>?), actionIndex)
    }; separator="\n">
    default: break
    }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
override <accessLevelOpenOK(parser)>
func sempred(_ _localctx: RuleContext?, _ ruleIndex: Int, _ predIndex: Int) throws -> Bool {

```

```

switch (ruleIndex) {
  <recog.sempredFuncs.values:{f|
case <f.ruleIndex>:
  return try <f.name>_sempred(_localctx?.castdown(<f.ctxType>.self), predIndex)); separator="\n">
  default: return true
  }
  <!return true;!>
}
<sempredFuncs.values; separator="\n">
<endif>
>>

```

```

parser_ctor(p) ::= <<

```

```

override <accessLevelOpenOK(parser)>
func getVocabulary() -> Vocabulary {
  return <p.name>.VOCABULARY
}

```

```

override <accessLevelNotOpen(parser)>
init(_ input:TokenStream) throws {
  RuntimeMetaData.checkVersion("4.7.2", RuntimeMetaData.VERSION)
  try super.init(input)
  _interp = ParserATNSimulator(self,<p.name>._ATN,<p.name>._decisionToDFA,
<p.name>._sharedContextCache)
}

```

```

>>

```

```

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleActionFunction(r, actions) ::= <<
private func <r.name>_action(_ localctx: <r.ctxType>?, _ actionIndex: Int) {
  switch (actionIndex) {
    <actions:{index|
case <index>:
  <actions.(index)>
  }; separator="\n">
  default: break
  }
}
}
>>

```

```

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleSempredFunction(r, actions) ::= <<

```

```

private func <r.name>_sempred(_ _localctx: <r.ctxType>!, _ predIndex: Int) throws -> Bool {
  switch (predIndex) {
  <actions: {index|
    case <index>:return <actions.(index)>; separator="\n">
    default: return true
  }
  <!return true;!>
}
>>

```

```

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<
<ruleCtx>
<altLabelCtxs: {l | <altLabelCtxs.(l)>; separator="\n">
@discardableResult
<if(currentRule.modifiers)><currentRule.modifiers: {f | <f> }><else> <accessLevelOpenOK(parser)> func
<endif><currentRule.name><(if(first(args))>_ <endif><args; separator=", _">) throws -> <currentRule.ctxType> {
  var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, getState())<currentRule.args: {a | ,
  <a.name>}>>)
  try enterRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>)
  <namedActions.init>
  <locals; separator="\n">
  defer {
    <finallyAction>
    try! exitRule()
  }
  do {
  <if(currentRule.hasLookaheadBlock)>
  var _alt:Int
  <endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  catch ANTLRException.recognition(let re) {
    _localctx.exception = re
    _errHandler.reportError(self, re)
    try _errHandler.recover(self, re)
  }
  <endif>

  return _localctx
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else> <accessLevelNotOpen(parser)> final <endif>
func <currentRule.name>(<if(first(args))>_ <endif><args; separator=", _">) throws -> <currentRule.ctxType> {
    return try <currentRule.name>(0<currentRule.args:{a | , <a.name>}>)
}
@discardableResult
private func <currentRule.name>(_ p<args:{a | , <a>}>: Int) throws -> <currentRule.ctxType> {
    let _parentctx: ParserRuleContext? = _ctx
    var _parentState: Int = getState()
    var _localctx: <currentRule.ctxType> = <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
<a.name>}>)
    var _prevctx: <currentRule.ctxType> = _localctx
    var _startState: Int = <currentRule.startState>
    try enterRecursionRule(_localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p)
    <namedActions.init>
    <locals; separator="\n">
    defer {
        <finallyAction>
        try! unrollRecursionContexts(_parentctx)
    }
    do {
    <if(currentRule.hasLookaheadBlock)>
        var _alt: Int
    <endif>
    <code>
    <postamble; separator="\n">
    <namedActions.after>
    }
    catch ANTLRException.recognition(let re) {
        _localctx.exception = re
        _errHandler.reportError(self, re)
        try _errHandler.recover(self, re)
    }

    return _localctx;
}
>>

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
try enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>)

```

```
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
```

```
>>
```

```
CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
```

```
<!/{!>
```

```
<locals; separator="\n">
```

```
<preamble; separator="\n">
```

```
<ops; separator="\n">
```

```
<!/{}!>
```

```
>>
```

```
LL1AltBlock(choice, preamble, alts, error) ::= <<
```

```
setState(<choice.stateNumber>)
```

```
try _errHandler.sync(self)
```

```
<if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
```

```
<preamble; separator="\n">
```

```
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))! {
```

```
<choice.altLook,alts:{look,alt | <cases(ttypes=look)>
```

```
<alt>
```

```
break }; separator="\n">
```

```
default:
```

```
<error>
```

```
}
```

```
>>
```

```
LL1OptionalBlock(choice, alts, error) ::= <<
```

```
setState(<choice.stateNumber>)
```

```
try _errHandler.sync(self)
```

```
switch (<parser.name>.Tokens(rawValue: try _input.LA(1))! {
```

```
<choice.altLook,alts:{look,alt | <cases(ttypes=look)>
```

```
<alt>
```

```
break }; separator="\n">
```

```
default:
```

```
break
```

```
}
```

```
>>
```

```
LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
```

```
setState(<choice.stateNumber>)
```

```
try _errHandler.sync(self)
```

```
<preamble; separator="\n">
```

```
if (<expr> {
```

```
<alts; separator="\n">
```

```
}
```

```
<!else if ( !(<followExpr> ) <error>!>
```

```
>>
```

```
LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
```

```

setState(<choice.stateNumber>)
try _errHandler.sync(self)
<preamble; separator="\n">
while (<loopExpr>) {
  <alts; separator="\n">
  setState(<choice.loopBackStateNumber>)
  try _errHandler.sync(self)
  <iteration>
}
>>

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.blockStartStateNumber>) <! alt block decision !>
try _errHandler.sync(self)
<preamble; separator="\n">
repeat {
  <alts; separator="\n">
  setState(<choice.stateNumber>); <! loopback/exit decision !>
  try _errHandler.sync(self)
  <iteration>
} while (<loopExpr>)
>>

// LL(*) stuff

AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = try _input.LT(1)<endif>
<preamble; separator="\n">
switch(try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)) {
  <alts:{alt |
case <i>:
  <alt>
  break }; separator="\n">
default: break
}
>>

OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>)
try _errHandler.sync(self)
switch (try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)) {
  <alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
  break }; separator="\n">
default: break

```

```
}  
>>
```

```
StarBlock(choice, alts, sync, iteration) ::= <<  
  setState(<choice.stateNumber>)  
  try _errHandler.sync(self)  
  _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)  
  while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER) {  
    if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {  
      <iteration>  
      <alts> <! should only be one !>  
    }  
    setState(<choice.loopBackStateNumber>)  
    try _errHandler.sync(self)  
    _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)  
  }  
>>
```

```
PlusBlock(choice, alts, error) ::= <<  
  setState(<choice.blockStartStateNumber>); <! alt block decision !>  
  try _errHandler.sync(self)  
  _alt = 1<if(!choice.ast.greedy)>+1<endif>;  
  repeat {  
    switch (_alt) {  
      <alts:{alt|  
        case <i><if(!choice.ast.greedy)>+1<endif>:  
          <alt>  
          break }; separator="\n">  
      default:  
        <error>  
    }  
    setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>  
    try _errHandler.sync(self)  
    _alt = try getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx)  
  } while (_alt != <choice.exitAlt> && _alt != ATN.INVALID_ALT_NUMBER)  
>>
```

```
Sync(s) ::= "sync(<s.expecting.name>);"
```

```
ThrowNoViableAlt(t) ::= "throw ANTLRException.recognition(e: NoViableAltException(self))"
```

```
TestSetInline(s) ::= <<  
<!<s.bitsets:{bits | <if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,  
bits)><endif>}; separator=" || ">!>  
//closure  
{ () -> Bool in  
  <if(rest(s.bitsets)>>var<else>let<endif> testSet: Bool = <first(s.bitsets):{bits |  
<if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s, bits)><endif>}&>
```

```

    <rest(s.bitsets): {bits | testSet = testSet || <if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s,
bits)><else><bitsetInlineComparison(s, bits)><endif>}; separator="\n">
    return testSet
}()
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <<
<!(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1 \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes: {ttype | (1 \<< <offsetShift(ttype, bits.shift)>}); separator=" | ">)) != 0)!>
{ () -> Bool in
    <! let test: Bool = (<testShiftInRange({<offsetShift(s.varName, bits.shift)>})>)!>
    <!var temp: Int64 = Int64(<offsetShift(s.varName, bits.shift)>)!>
    <!temp = (temp \< 0) ? (64 + (temp % 64)) : (temp % 64)!>
    <!let test1: Int64 = (Int64(1) \<< temp)!>
    <!var test2: Int64 = (<first(bits.ttypes): {ttype | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)}>)!>
    <!<rest(bits.ttypes): {ttype | test2 = test2 | Utils.bitLeftShift(<offsetShift(parserName(ttype), bits.shift)>)}>};
separator="\n">!>
    let testArray: [Int] = [<s.varName>, <bits.ttypes: {ttype | <parserName(ttype)>}; separator=", ">]
    <!var test2: Int64 = Utils.testBitLeftShiftArray(testArray)!>
    return Utils.testBitLeftShiftArray(testArray, <bits.shift>)
}()
>>

isZero ::= [
"0": true,
default: false
]
parserName(ttype) ::= <%
<parser.name>.Tokens.<ttype>.rawValue
%>
offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes: {ttype | <s.varName> == <parser.name>.Tokens.<ttype>.rawValue}; separator=" || ">
%>

cases(ttypes) ::= <<
<trunc(ttypes): {t | case .<t>:fallthrough} ; separator="\n">
<last(ttypes): {t | case .<t>:} ; separator="\n">

```


>>

```
InvokeRule(r, argExprsChunks) ::= <<
  setState(<r.stateNumber>)
  <if(r.labels)>
  try {
    let assignmentValue = try
  <r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)
  <r.labels:{1 | <labelref(l)> = assignmentValue} ; separator="\n">
    }()
  <else>try
  <r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>)<endif>
  >>
```

```
MatchToken(m) ::= <<
  setState(<m.stateNumber>)
  <if(m.labels)>
  try {
    let assignmentValue = try match(<parser.name>.Tokens.<m.name>.rawValue)
  <m.labels:{1 | <labelref(l)> = assignmentValue} ; separator="\n">
    }()
  <else>try match(<parser.name>.Tokens.<m.name>.rawValue)<endif>
  >>
```

```
MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"
```

```
MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"
```

```
CommonSetStuff(m, expr, capture, invert) ::= <<
  setState(<m.stateNumber>)
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }>try _input.LT(1)<endif>
  <capture>
  if (<if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr>)) {
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>try _errHandler.recoverInline(self)<if(m.labels)> as
  Token<endif>
  }
  else {
  _errHandler.reportMatch(self)
  try consume()
  }
  >>
```

```
Wildcard(w) ::= <<
  setState(<w.stateNumber>)
  <if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>try matchWildcard();
  >>
```

```
// ACTION STUFF
```

```

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
setState(<p.stateNumber>)
if (!(<chunks>)) {
    throw ANTLRException.recognition(e:FailedPredicateException(self, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>))
}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"
LexerMoreCommand() ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg)    ::= "_type = <arg>"
LexerChannelCommand(arg) ::= "_channel = <arg>"
LexerModeCommand(arg)    ::= "_mode = <arg>"
LexerPushModeCommand(arg) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValRef(a) ::= "_localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getText()! : \"\""
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getType() : 0)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getLine() : 0)"

```

```

TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getCharPositionInLine() : 0)"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getChannel() : 0)"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label> != nil ? <ctx(t)>.<t.label>!.getTokenIndex() : 0)"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label> != nil ? Int(<ctx(t)>.<t.label>!.getText()) : 0)"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.start?.description ?? \"\") : \"\""
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label> != nil ? (<ctx(r)>.<r.label>!.stop?.description ?? \"\") : \"\""
RulePropertyRef_text(r) ::= "<ctx(r)>.<r.label> != nil ? try
_input.getText(<ctx(r)>.<r.label>!.start,<ctx(r)>.<r.label>!.stop) : \"\""
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "_localctx.start"
ThisRulePropertyRef_stop(r) ::= "_localctx.stop"
ThisRulePropertyRef_text(r) ::= "(try _input.getText(_localctx.start, try _input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "_localctx"
ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::=
"(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "<t.name>: <SwiftTypeMap.(TokenLabelType())>!"
TokenTypeDecl(t) ::= "var <t.name>: Int = 0"
TokenListDecl(t) ::= "<t.name>: [Token] = [Token]()"
RuleContextDecl(r) ::= "<r.name>: <r.ctxName>!"
RuleContextListDecl(rdecl) ::= "<rdecl.name>: [<rdecl.ctxName>] = [<rdecl.ctxName>]()"

ContextTokenGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>() -> TerminalNode? {
return getToken(<parser.name>.Tokens.<t.name>.rawValue, 0)
}
>>

ContextTokenListGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>() -> [TerminalNode] {
return getTokens(<parser.name>.Tokens.<t.name>.rawValue)
}
>>

ContextTokenListIndexedGetterDecl(t) ::= <<
<accessLevelOpenOK(parser)>
func <t.name>(_ i:Int) -> TerminalNode? {
return getToken(<parser.name>.Tokens.<t.name>.rawValue, i)
}
>>

```

```

ContextRuleGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>() -> <r.ctxName>? {
    return getRuleContext(<r.ctxName>.self, 0)
  }
>>

ContextRuleListGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>() -> [<r.ctxName>] {
    return getRuleContexts(<r.ctxName>.self)
  }
>>

ContextRuleListIndexedGetterDecl(r) ::= <<
  <accessLevelOpenOK(parser)>
  func <r.name>(_ i: Int) -> <r.ctxName>? {
    return getRuleContext(<r.ctxName>.self, i)
  }
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = try _input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = try _input.LA(1)"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers,
  superClass={ParserRuleContext}) ::= <<

  <accessLevelNotOpen(parser)> class <struct.name>:
  <if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces>
  separator=", "><endif> {
  <attrs:{a | <accessLevelOpenOK(parser)> var <a>}; separator="\n">
  <getters:{g | <g>}; separator="\n">
  <! <if(ctorAttrs)> <accessLevelNotOpen(parser)> init(_ parent: ParserRuleContext,_ invokingState: Int) {
  super.init(parent, invokingState) }<endif> !>
  <if(ctorAttrs)>
  <accessLevelNotOpen(parser)> convenience init(_ parent: ParserRuleContext?, _ invokingState: Int<ctorAttrs:{a | ,
  _ <a>}> ) {
  self.init(parent, invokingState)

```

```

    <struct.ctorAttrs:{ a | self.<a.name> = <a.name>; separator="\n">
  }
<endif>

```

```

override <accessLevelOpenOK(parser)>
func getRuleIndex() -> Int {
  return <parser.name>.RULE_<struct.derivedFromName>
}
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
<! <accessLevelNotOpen(parser)> init() { }!>
<accessLevelOpenOK(parser)>
func copyFrom(_ ctx: <struct.name>) {
  super.copyFrom(ctx)
  <struct.attrs:{ a | self.<a.name> = ctx.<a.name>; separator="\n">
}
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">
}
>>

```

```

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
<accessLevelNotOpen(parser)> class <struct.name>: <currentRule.name; format="cap">Context {
  <attrs:{ a | <accessLevelNotOpen(parser)> var <a>; separator="\n">
  <getters:{ g | <g>; separator="\n">

```

```

  <accessLevelNotOpen(parser)>
  init(_ ctx: <currentRule.name; format="cap">Context) {
    super.init()
    copyFrom(ctx)
  }
  <dispatchMethods; separator="\n">
}
>>

```

```

ListenerDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func <if(method.isEnter)>enter<else>exit<endif>Rule(_ listener: ParseTreeListener) {
  if let listener = listener as? <parser.grammarName>Listener {
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)
  }
}
>>

```

```

VisitorDispatchMethod(method) ::= <<
override <accessLevelOpenOK(parser)>
func accept<T>(_ visitor: ParseTreeVisitor<T>) -> T? {
  if let visitor = visitor as? <parser.grammarName>Visitor {

```

```

    return visitor.visit<struct.derivedFromName; format="cap">(self)
  }
  else if let visitor = visitor as? <parser.grammarName>BaseVisitor {
    return visitor.visit<struct.derivedFromName; format="cap">(self)
  }
  else {
    return visitor.visitChildren(self)
  }
}
>>

AttributeDecl(d) ::= "<d.name>: <SwiftTypeMap.(d.type)><if(d.initValue)> = <d.initValue><else>!<endif>"

/** If we don't know location of label def x, use this template (_localctx as! <x.ctx.name> */
labelref(x) ::= "<if(!x.isLocal)>_localctx.castdown(<x.ctx.name>.self).<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "_localctx.castdown(<actionChunk.ctx.name>.self)"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "precpred(_ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "_ctx!.stop = try _input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
  _localctx = <ctxName>Context(_parentctx, _parentState);
  <if(label)>
  <if(isListLabel)>
  _localctx.<label>.append(_prevctx)
  <else>
  _localctx.<label> = _prevctx
  <endif>
  <endif>
  <if(label)>_localctx.<label> = _prevctx;<endif>
  try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
  _localctx = <currentAltLabel; format="cap">Context( <ruleName; format="cap">Context(_parentctx,
  _parentState))
  <if(label)>
  <if(isListLabel)>
  (_localctx as! <currentAltLabel; format="cap">Context).<label>.append(_prevctx)
  <else>
  (_localctx as! <currentAltLabel; format="cap">Context).<label> = _prevctx
  <endif>
  <endif>
  try pushNewRecursionContext(_localctx, _startState, <parser.name>.RULE_<ruleName>)

```

```
>>
```

```
recRuleReplaceContext(ctxName) ::= <<  
  _localctx = <ctxName>Context(_localctx)  
  _ctx = _localctx  
  _prevctx = _localctx  
>>
```

```
recRuleSetPrevCtx() ::= <<  
  if _parseListeners != nil {  
    try triggerExitRuleEvent()  
  }  
  _prevctx = _localctx  
>>
```

```
LexerFile(lexerFile, lexer, namedActions) ::= <<  
  <fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>  
  <if(lexerFile.genPackage)>  
  <!package <lexerFile.genPackage>;!>  
  <endif>  
  <namedActions.header>  
  import Antlr4
```

```
<lexer>  
>>
```

```
Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<  
  <accessLevelOpenOK(lexer)> class <lexer.name>: <superClass; null="Lexer"> {
```

```
  internal static var _decisionToDFA: [DFA] = {  
    var decisionToDFA = [DFA]()  
    let length = <lexer.name>._ATN.getNumberOfDecisions()  
    for i in 0..<length {  
      <! decisionToDFA[i] = DFA(<lexer.name>._ATN.getDecisionState(i)!, i);!>  
      decisionToDFA.append(DFA(<lexer.name>._ATN.getDecisionState(i)!, i))  
    }  
    return decisionToDFA  
  }()
```

```
  internal static let _sharedContextCache = PredictionContextCache()
```

```
  <accessLevelNotOpen(lexer)>  
  static let <lexer.tokens>:{k | <k>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>
```

```
  <if(lexer.channels)>  
  <accessLevelNotOpen(lexer)>  
  static let <lexer.channels>:{k | <k>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>  
  <endif>
```

```

<if(rest(lexer.modes))>
<accessLevelNotOpen(lexer)>
static let <rest(lexer.modes):{m| <m>=<i>}; separator=", ", wrap, anchor>
<endif>
<accessLevelNotOpen(lexer)>
static let channelNames: [String] = [
"DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
]

<accessLevelNotOpen(lexer)>
static let modeNames: [String] = [
<lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
]

<accessLevelNotOpen(lexer)>
static let ruleNames: [String] = [
<lexer.ruleNames:{r| "<r>"}; separator=", ", wrap, anchor>
]

<vocabulary(lexer.literalNames, lexer.symbolicNames,
accessLevelNotOpen(lexer))>

<namedActions.members>

override <accessLevelOpenOK(lexer)>
func getVocabulary() -> Vocabulary {
return <lexer.name>.VOCABULARY
}

<accessLevelNotOpen(lexer)>
required init(_ input: CharStream) {
RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION)
super.init(input)
_interp = LexerATNSimulator(self, <lexer.name>._ATN, <lexer.name>._decisionToDFA,
<lexer.name>._sharedContextCache)
}

override <accessLevelOpenOK(lexer)>
func getGrammarFileName() -> String { return "<lexer.grammarFileName>" }

override <accessLevelOpenOK(lexer)>
func getRuleNames() -> [String] { return <lexer.name>.ruleNames }

override <accessLevelOpenOK(lexer)>
func getSerializedATN() -> String { return <lexer.name>._serializedATN }

override <accessLevelOpenOK(lexer)>

```



```

func getChannelNames() -> [String] { return <lexer.name>.channelNames }

override <accessLevelOpenOK(lexer)>
func getModeNames() -> [String] { return <lexer.name>.modeNames }

override <accessLevelOpenOK(lexer)>
func getATN() -> ATN { return <lexer.name>._ATN }

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>

<accessLevelNotOpen(lexer)>
static let _serializedATN: String = <lexer.name>ATN().jsonString

<accessLevelNotOpen(lexer)>
static let _ATN: ATN = ATNDeserializer().deserializeFromJson(_serializedATN)
}
>>

/** Don't need to define anything. The tool generates a XParserATN.swift file (and same for lexer)
 * which is referenced from static field _serializedATN. This json string is passed to
 * deserializeFromJson(). Note this is not the "serialization as array of ints" that other targets
 * do. It is more or less the output of ATNPrinter which gets read back in.
 */
SerializedATN(model) ::= <<
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<SwiftTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".swift"

Found in path(s):
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/codegen/Swift/Swift.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD licence"]
Copyright (c) 2006 Kay Roepke
All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions

are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the format that mimicks GCC output.

*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<text> [error <id>]"

report(location, message, type) ::= "<location> <type>: <message>"

wantsSingleLineMessage() ::= "true"

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/templates/messages/formats/gnu.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2015 Dan McLaughlin, Mike Lischke

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/codegen/Cpp/Cpp.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

```

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** How to generate rules derived from left-recursive rules.

```

```

* These rely on recRuleAltPredicate(),
* recRuleArg(), recRuleSetResultAction(), recRuleSetReturnAction()
* templates in main language.stg
*/

```

```

group LeftRecursiveRules;

```

```

recRule(ruleName, argName, primaryAlts, opAlts, setResultAction,
        userRetvals, leftRecursiveRuleRefLabels) ::=

```

```

<<
<ruleName><if(userRetvals)> returns [<userRetvals>]<endif>
: ( { } <primaryAlts:{ alt | <alt.altText> }; separator="\n    | ">
  )
  (
    <opAlts; separator="\n    | ">
  )*
;
>>

```

```

recRuleAlt(alt, precOption, opPrec, pred) ::= <<
{<pred>}?<<precOption>=<opPrec>> <alt.altText>
>>

```

Found in path(s):

```

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/LeftRecursiveRules.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

```

[The "BSD licence"]

```

```

Copyright (c) 2006 Kay Roepke

```

```

All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the default format ANTLR uses.

*/

```
location(file, line, column) ::= "<file>(<line>,<column>)"
```

```
message(id, text) ::= "error <id> : <text>"
```

```
report(location, message, type) ::= "<location> : <type> <message.id> : <message.text>"
```

```
wantsSingleLineMessage() ::= "true"
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/templates/messages/formats/vs2005.stg
```

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

/** Find left-recursive rules */

tree grammar LeftRecursiveRuleWalker;

```
options {
  tokenVocab=ANTLRParser;
  ASTLabelType=GrammarAST;
}
```

```
@header {
package org.antlr.v4.parse;
```

```
import org.antlr.v4.misc.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
}
```

```
@members {
private String ruleName;
private int currentOuterAltNumber; // which outer alt of rule?
public int numAlts; // how many alts for this rule total?
```

```
public void setAltAssoc(AltAST altTree, int alt) {}
public void binaryAlt(AltAST altTree, int alt) {}
public void prefixAlt(AltAST altTree, int alt) {}
public void suffixAlt(AltAST altTree, int alt) {}
public void otherAlt(AltAST altTree, int alt) {}
public void setReturnValues(GrammarAST t) {}
}
```

```
@rulecatch { }
```

```
// TODO: can get parser errors for not matching pattern; make them go away
```

```
public
```

```

rec_rule returns [boolean isLeftRec]
@init
{
    currentOuterAltNumber = 1;
}
: ^( r=RULE id=RULE_REF {ruleName=$id.getText();}
    ruleModifier?
// (ARG_ACTION)? shouldn't allow args, right?
    (^ (RETURNS a=ARG_ACTION {setReturnValues($a);}))?
// (^(THROWS .+)? don't allow
    (^ (LOCALS ARG_ACTION) )? // TODO: copy these to gen'd code
    (^ (OPTIONS .*)
    | ^ (AT ID ACTION) // TODO: copy
    )*)
ruleBlock { $isLeftRec = $ruleBlock.isLeftRec;}
exceptionGroup
)
;

exceptionGroup
: exceptionHandler* finallyClause?
;

exceptionHandler
: ^ (CATCH ARG_ACTION ACTION)
;

finallyClause
: ^ (FINALLY ACTION)
;

ruleModifier
: PUBLIC
| PRIVATE
| PROTECTED
;

ruleBlock returns [boolean isLeftRec]
@init{boolean lr=false; this.numAlts = $start.getChildCount();}
: ^ ( BLOCK
    (
        o=outerAlternative
        {if ($o.isLeftRec) $isLeftRec = true;}
        {currentOuterAltNumber++;}
    )+
)
;

```

```

/** An alt is either prefix, suffix, binary, or ternary operation or "other" */
outerAlternative returns [boolean isLeftRec]
: (binary)=>      binary
                  {binaryAlt((AltAST)$start, currentOuterAltNumber); $isLeftRec=true;}
| (prefix)=>     prefix
                  {prefixAlt((AltAST)$start, currentOuterAltNumber);}
| (suffix)=>     suffix
                  {suffixAlt((AltAST)$start, currentOuterAltNumber); $isLeftRec=true;}
| nonLeftRecur   {otherAlt((AltAST)$start, currentOuterAltNumber);}
;

binary
: ^( ALT elementOptions? recurse element* recurse epsilonElement* )
  {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
;

prefix
: ^( ALT elementOptions?
  element+
  recurse epsilonElement*
)
  {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
;

suffix
: ^( ALT elementOptions? recurse element+ )
  {setAltAssoc((AltAST)$ALT,currentOuterAltNumber);}
;

nonLeftRecur
: ^(ALT elementOptions? element+)
;

recurse
: ^(ASSIGN ID recurseNoLabel)
| ^(PLUS_ASSIGN ID recurseNoLabel)
| recurseNoLabel
;

recurseNoLabel : {((CommonTree)input.LT(1)).getText().equals(ruleName)}? RULE_REF;

token returns [GrammarAST t=null]
: ^(ASSIGN ID s=token {$t = $s.t;})
| ^(PLUS_ASSIGN ID s=token {$t = $s.t;})
| b=STRING_LITERAL    {$t = $b;}
| ^(b=STRING_LITERAL elementOptions) {$t = $b;}
| ^(c=TOKEN_REF elementOptions) {$t = $c;}
| c=TOKEN_REF        {$t = $c;}

```


;

elementOptions

: ^(ELEMENT_OPTIONS elementOption*)

;

elementOption

: ID

| ^(ASSIGN ID ID)

| ^(ASSIGN ID STRING_LITERAL)

| ^(ASSIGN ID ACTION)

| ^(ASSIGN ID INT)

;

element

: atom

| ^(NOT element)

| ^(RANGE atom atom)

| ^(ASSIGN ID element)

| ^(PLUS_ASSIGN ID element)

| ^(SET setElement+)

| RULE_REF

| ebnf

| epsilonElement

;

epsilonElement

: ACTION

| SEMPRED

| EPSILON

| ^(ACTION elementOptions)

| ^(SEMPRED elementOptions)

;

setElement

: ^(STRING_LITERAL elementOptions)

| ^(TOKEN_REF elementOptions)

| STRING_LITERAL

| TOKEN_REF

;

ebnf: block

| ^(OPTIONAL block)

| ^(CLOSURE block)

| ^(POSITIVE_CLOSURE block)

;

block

: ^(BLOCK ACTION? alternative+)

;

alternative

: ^(ALT elementOptions? element+)

;

atom

: ^(RULE_REF ARG_ACTION? elementOptions?)

| ^(STRING_LITERAL elementOptions)

| STRING_LITERAL

| ^(TOKEN_REF elementOptions)

| TOKEN_REF

| ^(WILDCARD elementOptions)

| WILDCARD

| ^(DOT ID element)

;

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/LeftRecursiveRuleWalker.g

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* Copyright (c) 2014 Eric Vergnaud

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

```

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** ANTLR tool checks output templates are compatible with tool code generation.
* For now, a simple string match used on x.y of x.y.z scheme.
* Must match Tool.VERSION during load to templates.
*
* REQUIRED.
*/

```

```

javascriptTypeInitMap ::= [
  "bool":"false",
  "int":"0",
  "float":"0.0",
  "str":"",
  default:"{}" // anything other than a primitive type is an object
]

```

```

// args must be <object-model-object>, <fields-resulting-in-STs>

```

```

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
var antlr4 = require('antlr4/index');
<if(file.genListener)>
var <file.grammarName>Listener = require('./<file.grammarName>Listener').<file.grammarName>Listener;
<endif>
<if(file.genVisitor)>
var <file.grammarName>Visitor = require('./<file.grammarName>Visitor').<file.grammarName>Visitor;
<endif>

<namedActions.header>
<parser>
>>

```

```

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
var antlr4 = require('antlr4/index');

```

```

// This class defines a complete listener for a parse tree produced by <file.parserName>.

```

```

function <file.grammarName>Listener() {
  antlr4.tree.ParseTreeListener.call(this);
  return this;
}

```

```

<file.grammarName>Listener.prototype = Object.create(antlr4.tree.ParseTreeListener.prototype);

```

```

<file.grammarName>Listener.prototype.constructor = <file.grammarName>Listener;

<file.listenerNames:{lname |
// Enter a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Listener.prototype.enter<lname; format="cap"> = function(ctx) {
\};

// Exit a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Listener.prototype.exit<lname; format="cap"> = function(ctx) {
\};

}; separator="\n">

exports.<file.grammarName>Listener = <file.grammarName>Listener;
>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
var antlr4 = require('antlr4/index');

// This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

function <file.grammarName>Visitor() {
  antlr4.tree.ParseTreeVisitor.call(this);
  return this;
}

<file.grammarName>Visitor.prototype = Object.create(antlr4.tree.ParseTreeVisitor.prototype);
<file.grammarName>Visitor.prototype.constructor = <file.grammarName>Visitor;

<file.visitorNames:{lname |
// Visit a parse tree produced by <file.parserName>#<lname>.
<file.grammarName>Visitor.prototype.visit<lname; format="cap"> = function(ctx) {
  return this.visitChildren(ctx);
\};

}; separator="\n">

exports.<file.grammarName>Visitor = <file.grammarName>Visitor;
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
// jshint ignore: start
>>

```

```

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<if(superClass)>
var <superClass> = require('./<superClass>').<superClass>;
<endif>

var grammarFileName = "<parser.grammarFileName; format='java-escape'>";

<atn>

var atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);

var decisionsToDFA = atn.decisionToState.map( function(ds, index) { return new antlr4.dfa.DFA(ds, index); });

var sharedContextCache = new antlr4.PredictionContextCache();

var literalNames = [ <parser.literalNames: {t | <t>} ; null="null", separator=", ", wrap, anchor > ];

var symbolicNames = [ <parser.symbolicNames: {t | <t>} ; null="null", separator=", ", wrap, anchor > ];

var ruleNames = [ <parser.ruleNames: {r | "<r>"} ; separator=", ", wrap, anchor > ];

function <parser.name> (input) {
<superClass; null="antlr4.Parser">.call(this, input);
  this._interp = new antlr4.atn.ParserATNSimulator(this, atn, decisionsToDFA, sharedContextCache);
  this.ruleNames = ruleNames;
  this.literalNames = literalNames;
  this.symbolicNames = symbolicNames;
  <namedActions.members>
  return this;
}

<parser.name>.prototype = Object.create(<superClass; null="antlr4.Parser">.prototype);
<parser.name>.prototype.constructor = <parser.name>;

Object.defineProperty(<parser.name>.prototype, "atn", {
  get : function() {
    return atn;
  }
});

<parser.name>.EOF = antlr4.Token.EOF;
<if(parser.tokens)>
<parser.tokens: {k | <parser.name>.<k> = <parser.tokens.(k)>;} ; separator="\n", wrap, anchor >
<endif>

<parser.rules: {r | <parser.name>.RULE_<r.name> = <r.index>;} ; separator="\n", wrap, anchor >

<funcs; separator="\n">

```

```

<if(sempredFuncs)>
<parser.name>.prototype.sempred = function(localctx, ruleIndex, predIndex) {
switch(ruleIndex) {
<parser.sempredFuncs.values:{f | case <f.ruleIndex>:
return this.<f.name>_sempred(localctx, predIndex);}; separator="\n">
default:
    throw "No predicate with index:" + ruleIndex;
}
};

```

```

<sempredFuncs.values; separator="\n">
<endif>

```

```

exports.<parser.name> = <parser.name>;

```

```

>>

```

```

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
<lexer.name>.prototype.action = function(localctx, ruleIndex, actionIndex) {
switch (ruleIndex) {
<recog.actionFuncs.values:{f|
case <f.ruleIndex>:
this.<f.name>_action(localctx, actionIndex);
break;}; separator="\n">
default:
    throw "No registered action for:" + ruleIndex;
}
};

```

```

<actionFuncs.values; separator="\n">
<endif>

```

```

<if(sempredFuncs)>
<lexer.name>.prototype.sempred = function(localctx, ruleIndex, predIndex) {
switch (ruleIndex) {
<recog.sempredFuncs.values:{f| case <f.ruleIndex>:
return this.<f.name>_sempred(localctx, predIndex);}; separator="\n">
default:
    throw "No registered predicate for:" + ruleIndex;
}
};

```

```

<sempredFuncs.values; separator="\n">
<endif>

```

```

>>

```

```

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

<lexer.name>.prototype.<r.name>_action = function(localctx , actionIndex) {
  switch (actionIndex) {
    <actions:{index|
case <index>:
    <actions.(index)>
    break;}; separator="\n">
  default:
    throw "No registered action for:" + actionIndex;
  }
};
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
<if (r.factory.g.lexer)><lexer.name><else><parser.name><endif>.prototype.<r.name>_sempred = function(localctx,
predIndex) {
  switch(predIndex) {
    <actions:{index| case <index>:
return <actions.(index)>;}; separator="\n">
  default:
    throw "No predicate with index:" + predIndex;
  }
};
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<! Define fields of this parser to export the context classes !>
<parser.name>.<currentRule.ctxType> = <currentRule.ctxType>;

<parser.name>.prototype.<currentRule.name> = function(<currentRule.args:{a | <a.name>}; separator=", ">) {

  var localctx = new <currentRule.ctxType>(this, this._ctx, this.state<currentRule.args:{a | , <a.name>}>);
  this.enterRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>);
  <namedActions.init>

```

```

<locals; separator="\n">
try {
  <code>
  <postamble; separator="\n">
  <namedActions.after>
<if(exceptions)>
<exceptions; separator="\n">
<else>
} catch (re) {
if(re instanceof antlr4.error.RecognitionException) {
  localctx.exception = re;
  this._errHandler.reportError(this, re);
  this._errHandler.recover(this, re);
} else {
  throw re;
}
}<endif> finally {
  <finallyAction>
  this.exitRule();
}
return localctx;
};

```

>>

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
namedActions,finallyAction,postamble) ::=

```

<<

```

<ruleCtx>

```

```

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

<parser.name>.prototype.<currentRule.name> = function(_p<if(currentRule.args)>, <args:{a | , <a>}><endif>) {

```

```

  if(_p===undefined) {

```

```

    _p = 0;

```

```

  }

```

```

  var _parentctx = this._ctx;

```

```

  var _parentState = this.state;

```

```

  var localctx = new <currentRule.ctxType>(this, this._ctx, _parentState<args:{a | , <a.name>}>);

```

```

  var _prevctx = localctx;

```

```

  var _startState = <currentRule.startState>;

```

```

  this.enterRecursionRule(localctx, <currentRule.startState>, <parser.name>.RULE_<currentRule.name>, _p);

```

```

  <namedActions.init>

```

```

  <locals; separator="\n">

```

```

  try {

```

```

    <code>

```

```

    <postamble; separator="\n">

```

```

    <namedActions.after>

```



```

} catch( error) {
    if(error instanceof antlr4.error.RecognitionException) {
        localctx.exception = error;
        this._errHandler.reportError(this, error);
        this._errHandler.recover(this, error);
    } else {
        throw error;
    }
} finally {
    <finallyAction>
    this.unrollRecursionContexts(_parentctx)
}
return localctx;
};

```

>>

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(this, localctx);<endif>
this.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = this._input.LT(1);<endif>
<preamble; separator="\n">
switch(this._input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
    <alt>
    break;}; separator="\n">
default:
    <error>
}
>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
switch (this._input.LA(1)) {

```

```

<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  break;}; separator="\n">
default:
  break;
}
>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
if(<expr>) {
  <alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<preamble; separator="\n">
while(<loopExpr>) {
  <alts; separator="\n">
  this.state = <choice.loopBackStateNumber>;
  this._errHandler.sync(this);
  <iteration>
}
>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
<preamble; separator="\n">
do {
  <alts; separator="\n">
  this.state = <choice.stateNumber>; <! loopback/exit decision !>
  this._errHandler.sync(this);
  <iteration>
} while(<loopExpr>);
>>

```

```

// LL(*) stuff

```

```

AltBlock(choice, preamble, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>

```

```

<preamble; separator="\n">
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
switch(la_) {
<alts:{alt |
case <i>:
    <alt>
    break;
}; separator="\n">
}
>>

```

```

OptionalBlock(choice, alts, error) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var la_ = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
<alts:{alt |
if(la_===<i><if(!choice.ast.greedy)>+1<endif>) {
    <alt>
}; separator="\n} else ">
}
>>

```

```

StarBlock(choice, alts, sync, iteration) ::= <<
this.state = <choice.stateNumber>;
this._errHandler.sync(this);
var _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx)
while(_alt!=<choice.exitAlt> && _alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER) {
    if(_alt===1<if(!choice.ast.greedy)>+1<endif>) {
        <iteration>
        <alts> <! should only be one !>
    }
    this.state = <choice.loopBackStateNumber>;
    this._errHandler.sync(this);
    _alt = this._interp.adaptivePredict(this._input,<choice.decision>,this._ctx);
}
>>

```

```

PlusBlock(choice, alts, error) ::= <<
this.state = <choice.blockStartStateNumber>; <! alt block decision !>
this._errHandler.sync(this);
var _alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
    switch (_alt) {
    <alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
    break;}; separator="\n">

```

```

default:
  <error>
}
this.state = <choice.loopBackStateNumber>; <! loopback/exit decision !>
this._errHandler.sync(this);
_alt = this._interp.adaptivePredict(this._input,<choice.decision>, this._ctx);
} while ( _alt!=<choice.exitAlt> && _alt!=antlr4.atn.ATN.INVALID_ALT_NUMBER );
>>

Sync(s) ::= "sync(<s.expecting.name>)"

ThrowNoViableAlt(t) ::= "throw new antlr4.error.NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Javascript language spec - shift operators are 32 bits long max
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x1f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> && ((1 \<< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<< <offsetShiftType(ttype, bits.shift)>)}); separator=" | ">)) !== 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>===<parser.name>.<ttype>}; separator=" || ">
%>

cases(ttypes) ::= <<

```

```
<ttypes:{t | case <parser.name>.<t>:}; separator="\n">
```

```
>>
```

```
InvokeRule(r, argExprsChunks) ::= <<
```

```
this.state = <r.stateNumber>;
```

```
<if(r.labels)><r.labels:{l | <labelref(l)> =
```

```
>><endif>this.<r.name><if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)><endif><endif><argExprsChunks>;
```

```
>>
```

```
MatchToken(m) ::= <<
```

```
this.state = <m.stateNumber>;
```

```
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>this.match(<parser.name>.<m.name>);
```

```
>>
```

```
MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"
```

```
MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"
```

```
CommonSetStuff(m, expr, capture, invert) ::= <<
```

```
this.state = <m.stateNumber>;
```

```
<if(m.labels)><m.labels:{l | <labelref(l)> = }>this._input.LT(1);<endif>
```

```
<capture>
```

```
<if(invert)>if(<m.varName>\<=0 || <expr><else>if(!(<expr>))<endif> {
```

```
  <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>this._errHandler.recoverInline(this);
```

```
}
```

```
else {
```

```
  this._errHandler.reportMatch(this);
```

```
  this.consume();
```

```
}
```

```
>>
```

```
Wildcard(w) ::= <<
```

```
this.state = <w.stateNumber>;
```

```
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>this.matchWildcard();
```

```
>>
```

```
// ACTION STUFF
```

```
Action(a, foo, chunks) ::= "<chunks>"
```

```
ArgAction(a, chunks) ::= "<chunks>"
```

```
SemPred(p, chunks, failChunks) ::= <<
```

```
this.state = <p.stateNumber>;
```

```
if (!( <chunks>)) {
```

```
  throw new antlr4.error.FailedPredicateException(this, <p.predicate><if(failChunks)>,<
```

```
<failChunks><elseif(p.msg)>, <p.msg><endif>);
```

```

}
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
  <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "this.skip()"
LexerMoreCommand() ::= "this.more()"
LexerPopModeCommand() ::= "this.popMode()"
LexerTypeCommand(arg, grammar) ::= "this._type = <arg>"
LexerChannelCommand(arg, grammar) ::= "this._channel = <arg>"
LexerModeCommand(arg, grammar) ::= "this._mode = <arg>"
LexerPushModeCommand(arg, grammar) ::= "this.pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValRef(a) ::= "localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label>===null ? null : <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label>=== null ? 0 : parseInt(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>===null ? null : <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r) ::= "<ctx(r)>.<r.label>===null ? null : this._input.getText(new antlr4.Interval(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "this"

```

```

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r) ::= "localctx.stop"
ThisRulePropertyRef_text(r) ::= "this._input.getText(new antlr4.Interval(localctx.start, this._input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "localctx"
ThisRulePropertyRef_parser(r) ::= "this"

```

```

NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

```

```

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.push(<labelref(a.label)>);"

```

```

TokenDecl(t) ::= "this.<t.name> = null; // <TokenLabelType()>"
TokenTypeDecl(t) ::= "var <t.name> = 0; // <TokenLabelType()> type"
TokenListDecl(t) ::= "this.<t.name> = []; // of <TokenLabelType()>s"
RuleContextDecl(r) ::= "this.<r.name> = null; // <r.ctxName>"
RuleContextListDecl(rdecl) ::= "this.<rdecl.name> = []; // of <rdecl.ctxName>s"

```

```

ContextTokenGetterDecl(t) ::= <<
<t.name> = function() {
    return this.getToken(<parser.name>.<t.name>, 0);
};
>>

```

```

// should never be called
ContextTokenListGetterDecl(t) ::= <<
def <t.name>_list(self):
    return self.getTokens(<parser.name>.<t.name>)
>>

```

```

ContextTokenListIndexedGetterDecl(t) ::= <<
<t.name> = function(i) {
    if(i===undefined) {
        i = null;
    }
    if(i===null) {
        return this.getTokens(<parser.name>.<t.name>);
    } else {
        return this.getToken(<parser.name>.<t.name>, i);
    }
};
>>

```

```

ContextRuleGetterDecl(r) ::= <<
<r.name> = function() {
    return this.getTypedRuleContext(<r.ctxName>,0);
};

```

```

>>

// should never be called
ContextRuleListGetterDecl(r) ::= <<
def <r.name>_list(self):
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r) ::= <<
<r.name> = function(i) {
    if(i===undefined) {
        i = null;
    }
    if(i===null) {
        return this.getTypedRuleContexts(<r.ctxName>);
    } else {
        return this.getTypedRuleContext(<r.ctxName>,i);
    }
};
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = this._input.LA(1);"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<
function <struct.name>(parser, parent, invokingState<struct.ctorAttrs:{a | , <a.name>}>) {
    if(parent===undefined) {
        parent = null;
    }
    if(invokingState===undefined || invokingState===null) {
        invokingState = -1;
    }
    <if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.call(this, parent,
invokingState);
    this.parser = parser;

```



```

    this.ruleIndex = <parser.name>.RULE_<struct.derivedFromName>;
    <attrs:{ a | <a>}; separator="\n">
    <struct.ctorAttrs:{ a | this.<a.name> = <a.name> || null;}; separator="\n">
    return this;
}

<struct.name>.prototype =
Object.create(<if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.prototype);
<struct.name>.prototype.constructor = <struct.name>;

<getters:{ g | <struct.name>.prototype.<g>}; separator="\n\n">

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
<struct.name>.prototype.copyFrom = function(ctx) {
<if(contextSuperClass)><contextSuperClass><else>antlr4.ParserRuleContext<endif>.prototype.copyFrom.call(this,
ctx);
    <struct.attrs:{ a | this.<a.name> = ctx.<a.name>}; separator="\n">
};
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs,getters,dispatchMethods) ::= <<
function <struct.name>(parser, ctx) {
    <currentRule.name; format="cap">Context.call(this, parser);
    <attrs:{ a | <a>}; separator="\n">
    <currentRule.name; format="cap">Context.prototype.copyFrom.call(this, ctx);
    return this;
}

<struct.name>.prototype = Object.create(<currentRule.name; format="cap">Context.prototype);
<struct.name>.prototype.constructor = <struct.name>;

<! Define fields of this parser to export this struct/context class !>
<parser.name>.<struct.name> = <struct.name>;

<getters:{ g | <struct.name>.prototype.<g>}; separator="\n\n">
<dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
<struct.name>.prototype.<if(method.isEnter)>enter<else>exit<endif>Rule = function(listener) {
    if(listener instanceof <parser.grammarName>Listener ) {
        listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(this);
    }
}

```

```

};

>>

VisitorDispatchMethod(method) ::= <<
<struct.name>.prototype.accept = function(visitor) {
  if ( visitor instanceof <parser.grammarName>Visitor ) {
    return visitor.visit<struct.derivedFromName; format="cap">(this);
  } else {
    return visitor.visitChildren(this);
  }
};

>>

AttributeDecl(d) ::= "this.<d.name> = <if(d.initValue)><d.initValue><else>null<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "this.precpred(this._ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "this._ctx.stop = this._input.LT(-1);"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = new <ctxName>Context(this, _parentctx, _parentState);
<if(label)>localctx.<label> = _prevctx;<endif>
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = new <currentAltLabel; format="cap">Context(this, new <ruleName; format="cap">Context(this,
_parentctx, _parentState));
<if(label)>
<if(isListLabel)>
localctx.<label>.push(_prevctx);
<else>
localctx.<label> = _prevctx;
<endif>
<endif>
this.pushNewRecursionContext(localctx, _startState, <parser.name>.RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<

```

```

localctx = new <ctxName>Context(this, localctx);
this._ctx = localctx;
_prevctx = localctx;
>>

```

```

recRuleSetPrevCtx() ::= <<
if(this._parseListeners!==null) {
  this.triggerExitRuleEvent();
}
_prevctx = localctx;
>>

```

```

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
var antlr4 = require('antlr4/index');

```

```

<namedActions.header>

```

```

<lexer>

```

```

>>

```

```

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<

```

```

<if(superClass)>
var <superClass> = require('./<superClass>').<superClass>;
<endif>

```

```

<atn>

```

```

var atn = new antlr4.atn.ATNDeserializer().deserialize(serializedATN);

```

```

var decisionsToDFA = atn.decisionToState.map( function(ds, index) { return new antlr4.dfa.DFA(ds, index); });

```

```

function <lexer.name>(input) {
  <if(superClass)><superClass><else>antlr4.Lexer<endif>.call(this, input);
  this._interp = new antlr4.atn.LexerATNSimulator(this, atn, decisionsToDFA, new
antlr4.PredictionContextCache());
  return this;
}

```

```

<lexer.name>.prototype = Object.create(<if(superClass)><superClass><else>antlr4.Lexer<endif>.prototype);
<lexer.name>.prototype.constructor = <lexer.name>;

```

```

Object.defineProperty(<lexer.name>.prototype, "atn", {
  get : function() {
    return atn;
  }
}

```

```

});

<lexer.name>.EOF = antlr4.Token.EOF;
<lexer.tokens:{k | <lexer.name>.<k> = <lexer.tokens.(k)>;}; separator="\n", wrap, anchor>

<if(lexer.channels)>
<lexer.channels:{c | <lexer.name>.<c> = <lexer.channels.(c)>;}; separator="\n">

<endif>
<if(rest(lexer.modes))>
<rest(lexer.modes):{m | <lexer.name>.<m> = <i>;}; separator="\n">

<endif>
<lexer.name>.prototype.channelNames = [ "DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>,
<lexer.channels:{c | "<c>"}; separator=", ", wrap, anchor><endif> ];

<lexer.name>.prototype.modeNames = [ <lexer.modes:{m | "<m>"}; separator=", ", wrap, anchor> ];

<lexer.name>.prototype.literalNames = [ <lexer.literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor> ];

<lexer.name>.prototype.symbolicNames = [ <lexer.symbolicNames:{t | <t>}; null="null", separator=", ", wrap,
anchor> ];

<lexer.name>.prototype.ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ];

<lexer.name>.prototype.grammarFileName = "<lexer.grammarFileName>";

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

exports.<lexer.name> = <lexer.name>;

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

var serializedATN = ["<model.serialized; wrap={", <n> " }>"].join("");

>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<javacriptTypeInitMap.(typeName)>
>>

```

```
codeFileExtension() ::= ".js"
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-  
jar/org/antlr/v4/tool/templates/codegen/JavaScript/JavaScript.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2012-2016 Terence Parr
```

```
* Copyright (c) 2012-2016 Sam Harwell
```

```
* Copyright (c) 2014 Eric Vergnaud
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

```
* modification, are permitted provided that the following conditions
```

```
* are met:
```

```
*
```

```
* 1. Redistributions of source code must retain the above copyright
```

```
* notice, this list of conditions and the following disclaimer.
```

```
* 2. Redistributions in binary form must reproduce the above copyright
```

```
* notice, this list of conditions and the following disclaimer in the
```

```
* documentation and/or other materials provided with the distribution.
```

```
* 3. The name of the author may not be used to endorse or promote products
```

```
* derived from this software without specific prior written permission.
```

```
*
```

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```

```
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

```
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

```
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
```

```
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
```

```
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

```
/** ANTLR tool checks output templates are compatible with tool code generation.
```

```
* For now, a simple string match used on x.y of x.y.z scheme.
```

```
* Must match Tool.VERSION during load to templates.
```

```
*
```

```
* REQUIRED.
```

```
*/
```

```
pythonTypeInitMap ::= [
```

```
  "bool": "False",
```

```

    "int": "0",
    "float": "0.0",
    "str": "",
    default: "None" // anything other than a primitive type is an object
]

// args must be <object-model-object>, <fields-resulting-in-STs>

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
# encoding: utf-8
from __future__ import print_function
from antlr4 import *
from io import StringIO
import sys

<namedActions.header>
<parser>

>>

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
<header>

# This class defines a complete listener for a parse tree produced by <file.parserName>.
class <file.grammarName>Listener(ParseTreeListener):

    <file.listenerNames: {lname |
# Enter a parse tree produced by <file.parserName>#<lname>.
def enter<lname; format="cap">(self, ctx):
    pass

# Exit a parse tree produced by <file.parserName>#<lname>.
def exit<lname; format="cap">(self, ctx):
    pass

}; separator="\n">

>>

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
<header>

```

This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

```
class <file.grammarName>Visitor(ParseTreeVisitor):
```

```
    <file.visitorNames:{lname |
```

```
    # Visit a parse tree produced by <file.parserName>#<lname>.
```

```
    def visit<lname; format="cap">(self, ctx):
```

```
        return self.visitChildren(ctx)
```

```
}; separator="\n">
```

```
>>
```

```
fileHeader(grammarFileName, ANTLRVersion) ::= <<
```

```
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
```

```
>>
```

```
Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
```

```
<Parser_(ctor="parser_ctor", ...)>
```

```
>>
```

```
Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
```

```
<if(superClass)>
```

```
from .<superClass> import <superClass>
```

```
<endif>
```

```
<atn>
```

```
class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):
```

```
    grammarFileName = "<parser.grammarFileName>"
```

```
    atn = ATNDeserializer().deserialize(serializedATN())
```

```
    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]
```

```
    sharedContextCache = PredictionContextCache()
```

```
    literalNames = [ <parser.literalNames:{t | u<t>}; null="u\"<INVALID>\"", separator=", ", wrap, anchor ]
```

```
    symbolicNames = [ <parser.symbolicNames:{t | u<t>}; null="u\"<INVALID>\"", separator=", ", wrap, anchor ]
```

```
    <parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor>
```

```
    ruleNames = [ <parser.ruleNames:{r | u"<r>"}; separator=", ", wrap, anchor ]
```

```
    EOF = <TokenLabelType().EOF
```

```

<if(parser.tokens)>
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor>
<endif>

<parser:(ctor())>

<namedActions.members>

<funcs; separator="\n">

<if(sempredFuncs)>
def sempred(self, localctx, ruleIndex, predIndex):
    if self._predicates == None:
        self._predicates = dict()
<parser.sempredFuncs.values:{ f |
    self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n    ">
    pred = self._predicates.get(ruleIndex, None)
    if pred is None:
        raise Exception("No predicate with index:" + str(ruleIndex))
    else:
        return pred(localctx, predIndex)

<sempredFuncs.values; separator="\n">
<endif>

>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
def action(self, localctx, ruleIndex, actionIndex):
    if self._actions is None:
        actions = dict()
<recog.actionFuncs.values:{ f |
    actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
    self._actions = actions
    action = self._actions.get(ruleIndex, None)
    if action is not None:
        action(localctx, actionIndex)
    else:
        raise Exception("No registered action for:" + str(ruleIndex))

<actionFuncs.values; separator="\n">

<endif>
<if(sempredFuncs)>

```



```

def sempred(self, localctx, ruleIndex, predIndex):
    if self._predicates is None:
        preds = dict()
<recog.sempredFuncs.values:{f
    preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
    self._predicates = preds
    pred = self._predicates.get(ruleIndex, None)
    if pred is not None:
        return pred(localctx, predIndex)
    else:
        raise Exception("No registered predicate for:" + str(ruleIndex))

<sempredFuncs.values; separator="\n">
<endif>
>>

parser_ctor(p) ::= <<
def __init__(self, input, output=sys.stdout):
    super(<parser.name>, self).__init__(input, output=output)
    self.checkVersion("<file.ANTLRVersion>")
    self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
    self._predicates = None

>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<

def <r.name>_action(self, localctx , actionIndex):
<actions:{index|
<if(first(actions))>
    if actionIndex == <index>:
        <actions.(index)>
<elseif(rest(actions))>
    elif actionIndex == <index>:
        <actions.(index)>
<endif> }; separator="\n">
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
def <r.name>_sempred(self, localctx, predIndex):
    <actions:{index|
<if(first(actions))>

```

```

    if predIndex == <index>:
        return <actions.(index)>
<elseif(rest(actions))>
    elif predIndex == <index>:
        return <actions.(index)>
<endif> }; separator="\n">

>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self<currentRule.args:{a | , <a.name>}>):

    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args:{a | , <a.name>}>)
    self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
    <namedActions.init>
    <locals; separator="\n">
    try:
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    <if(exceptions)>
    <exceptions; separator="\n">
    <else>
    except RecognitionException as re:
        localctx.exception = re
        self._errHandler.reportError(self, re)
        self._errHandler.recover(self, re)
    <endif>
    finally:
        <finallyAction>
        self.exitRule()
    return localctx

>>

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
namedAction,finallyAction,postamble) ::=

<<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

def <currentRule.name>(self, _p=0<if(currentRule.args)>, <args:{a | , <a>}><endif>):
  _parentctx = self._ctx
  _parentState = self.state
  localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{a | , <a.name>}>)
  _prevctx = localctx
  _startState = <currentRule.startState>
  self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
  <namedActions.init>
  <locals; separator="\n">
  try:
    <code>
    <postamble; separator="\n">
    <namedActions.after>
  except RecognitionException as re:
    localctx.exception = re
    self._errHandler.reportError(self, re)
    self._errHandler.recover(self, re)
  finally:
    <finallyAction>
    self.unrollRecursionContexts(_parentctx)
  return localctx

>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel>
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\nel">
else:
  <error>

```

>>

```
LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
  <alt>
  pass}; separator="\n">
else:
  pass
>>
```

```
LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
  <alts; separator="\n">

<!else if ( !(<followExpr> ) <error>!>
>>
```

```
LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
  <alts; separator="\n">
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  <iteration>

>>
```

```
LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">
while True:
  <alts; separator="\n">
  self.state = <choice.stateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  <iteration>
if not (<loopExpr>):
  break
```

>>

// LL(*) stuff

```
AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i>:
    <alt>
    pass
}; separator="\nel">
```

>>

```
OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
    <alt>
}; separator="\nel">
```

>>

```
StarBlock(choice, alts, sync, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
    if _alt==1<if(!choice.ast.greedy)>+1<endif>:
        <iteration>
        <alts> <! should only be one !>
    self.state = <choice.loopBackStateNumber>
    self._errHandler.sync(self)
    _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
```

>>

```
PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
```

```

while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  <alts:{alt|
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">
  else:
    <error>
    self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
    self._errHandler.sync(self)
    _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

>>

Sync(s) ::= "sync(<s.expecting.name>)"

ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" or ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShiftVar(s.varName, bits.shift)>})> and ((1 \<< <offsetShiftVar(s.varName,
bits.shift)>) & (<bits.ttypes:{ttype | (1 \<< <offsetShiftType(ttype, bits.shift)>)); separator=" | ">)) != 0
%>

isZero ::= [
"0":true,
default:false
]

offsetShiftVar(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

offsetShiftType(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%

```

```

<bits.ttypes:{ ttype | <s.varName>===<parser.name>.<ttype>}; separator=" or ">
%>

cases(ttypes) ::= <<
if token in [<ttypes:{t | <parser.name>.<t>}; separator=", ">]:
>>

InvokeRule(r, argExprsChunks) ::= <<
self.state = <r.stateNumber>
<if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif>self.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)><endif><endif><argExprsChu
nks>)
>>

MatchToken(m) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>self.match(<parser.name>.<m.name>)
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{l | <labelref(l)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr>)<endif>:
<if(m.labels)><m.labels:{l | <labelref(l)> = }><else> <endif>self._errHandler.recoverInline(self)
else:
    self._errHandler.reportMatch(self)
    self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels:{l | <labelref(l)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:

```

```

    from antlr4.error.Errors import FailedPredicateException
    raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
    <p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"
LexerMoreCommand() ::= "more()"
LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg, grammar) ::= "_type = <arg>"
LexerChannelCommand(arg, grammar) ::= "_channel = <arg>"
LexerModeCommand(arg, grammar) ::= "_mode = <arg>"
LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "localctx.<a.name>"
LocalRef(a) ::= "localctx.<a.name>"
RetValRef(a) ::= "localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"
TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type)"
TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"
TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"
TokenPropertyRef_channel(t) ::= "(0 if (<ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"
TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"
TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"
RulePropertyRef_stop(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"
RulePropertyRef_text(r) ::= "(None if <ctx(r)>.<r.label> is None else

```



```

self._input.getText((<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"
ThisRulePropertyRef_stop(r) ::= "localctx.stop"
ThisRulePropertyRef_text(r) ::= "self._input.getText((localctx.start, self._input.LT(-1)))"
ThisRulePropertyRef_ctx(r) ::= "localctx"
ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"

TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"

ContextTokenGetterDecl(t) ::= <<
def <t.name>(self):
    return self.getToken(<parser.name>.<t.name>, 0)
>>

// should never be called
ContextTokenListGetterDecl(t) ::= <<
def <t.name>_list(self):
    return self.getTokens(<parser.name>.<t.name>)
>>

ContextTokenListIndexedGetterDecl(t) ::= <<
def <t.name>(self, i=None):
    if i is None:
        return self.getTokens(<parser.name>.<t.name>)
    else:
        return self.getToken(<parser.name>.<t.name>, i)
>>

ContextRuleGetterDecl(r) ::= <<
def <r.name>(self):
    return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)
>>

// should never be called
ContextRuleListGetterDecl(r) ::= <<

```

```

def <r.name>_list(self):
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)

>>

ContextRuleListIndexedGetterDecl(r) ::= <<
def <r.name>(self, i=None):
    if i is None:
        return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
    else:
        return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)

>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
 * r becomes rContext.
 */
RuleContextNameSuffix() ::= "Context"

ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
ImplicitSetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "<label>"

CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"

StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):

    def __init__(self, parser, parent=None, invokingState=-1<struct.ctorAttrs:{a | , <a.name>=None}>):
        super(<parser.name>.<struct.name>, self).__init__(parent, invokingState)
        self.parser = parser
        <attrs:{a | <a>}; separator="\n">
        <struct.ctorAttrs:{a | self.<a.name> = <a.name>}; separator="\n">

        <getters:{g | <g>}; separator="\n\n">

    def getRuleIndex(self):
        return <parser.name>.RULE_<struct.derivedFromName>

<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
    def copyFrom(self, ctx):
        super(<parser.name>.<struct.name>, self).copyFrom(ctx)
        <struct.attrs:{a | self.<a.name> = ctx.<a.name>}; separator="\n">

```

```

<endif>
  <dispatchMethods; separator="\n">
  <extensionMembers; separator="\n">

>>

AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
class <struct.name>( <currentRule.name; format="cap">Context):

  def __init__(self, parser, ctx): # actually a <parser.name>.<currentRule.name; format="cap">Context)
    super(<parser.name>.<struct.name>, self).__init__(parser)
    <attrs:{ a | <a> }; separator="\n">
    self.copyFrom(ctx)

  <getters:{ g | <g> }; separator="\n">

  <dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener):
  if hasattr(listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">"):
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor):
  if hasattr(visitor, "visit<struct.derivedFromName; format="cap">"):
    return visitor.visit<struct.derivedFromName; format="cap">(self)
  else:
    return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "self._ctx.stop = self._input.LT(-1)"

```

```

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

```

```

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName>;
format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

```

```

recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

```

```

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
    self.triggerExitRuleEvent()
_prevctx = localctx
>>

```

```

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
# encoding: utf-8
from __future__ import print_function
from antlr4 import *
from io import StringIO
import sys

<namedActions.header>

<lexer>
>>

```

```

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<

```

```

<if(superClass)>
from .<superClass> import <superClass>

<endif>

<atn>

class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):

    atn = ATNDeserializer().deserialize(serializedATN())

    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

<if(lexer.channels)>
    <lexer.channels:{c| <c> = <lexer.channels.(c)>}; separator="\n">

<endif>
<if(rest(lexer.modes)>
    <rest(lexer.modes):{m| <m> = <i>}; separator="\n">

<endif>
    <lexer.tokens:{k| <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>

    channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels:{c|
u"<c>"}; separator=", ", wrap, anchor><endif> ]

    modeNames = [ <lexer.modes:{m| u"<m>"}; separator=", ", wrap, anchor> ]

    literalNames = [ u"\<INVALID>",
        <lexer.literalNames:{t| u<t>}; separator=", ", wrap, anchor> ]

    symbolicNames = [ u"\<INVALID>",
        <lexer.symbolicNames:{t| u<t>}; separator=", ", wrap, anchor> ]

    ruleNames = [ <lexer.ruleNames:{r| u"<r>"}; separator=", ", wrap, anchor> ]

    grammarFileName = u"<lexer.grammarFileName>"

    def __init__(self, input=None, output=sys.stdout):
        super(<lexer.name>, self).__init__(input, output=output)
        self.checkVersion("<lexerFile.ANTLRVersion>")
        self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
        self._actions = None
        self._predicates = None

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

```

```
>>
```

```
SerializedATN(model) ::= <<  
<! only one segment, can be inlined !>
```

```
def serializedATN():  
    with StringIO() as buf:  
        buf.write(u"<model.serialized; wrap={ }<\n>    buf.write(u" }>")  
        return buf.getvalue()
```

```
>>
```

```
/** Using a type to init value map, try to init a type; if not in table  
 * must be an object, default value is "null".  
 */  
initValue(typeName) ::= <<  
<pythonTypeInitMap.(typeName)>  
>>
```

```
codeFileExtension() ::= ".py"
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-  
jar/org/antlr/v4/tool/templates/codegen/Python2/Python2.stg  
No license file was found, but licenses were detected in source scan.
```

```
/*  
 * [The "BSD license"]  
 * Copyright (c) 2012-2016 Terence Parr  
 * Copyright (c) 2012-2016 Sam Harwell  
 * All rights reserved.  
 *  
 * Redistribution and use in source and binary forms, with or without  
 * modification, are permitted provided that the following conditions  
 * are met:  
 *  
 * 1. Redistributions of source code must retain the above copyright  
 * notice, this list of conditions and the following disclaimer.  
 * 2. Redistributions in binary form must reproduce the above copyright  
 * notice, this list of conditions and the following disclaimer in the  
 * documentation and/or other materials provided with the distribution.  
 * 3. The name of the author may not be used to endorse or promote products  
 * derived from this software without specific prior written permission.  
 *  
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR  
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES  
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```
javaTypeInitMap ::= [  
  "int": "0",  
  "long": "0",  
  "float": "0.0f",  
  "double": "0.0",  
  "boolean": "false",  
  "byte": "0",  
  "short": "0",  
  "char": "0",  
  default: "null" // anything other than a primitive type is an object  
]
```

```
// args must be <object-model-object>, <fields-resulting-in-STs>
```

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
<if(file.genPackage)>  
package <file.genPackage>;  
<endif>  
<namedActions.header>  
import org.antlr.v4.runtime.atn.*;  
import org.antlr.v4.runtime.dfa.DFA;  
import org.antlr.v4.runtime.*;  
import org.antlr.v4.runtime.misc.*;  
import org.antlr.v4.runtime.tree.*;  
import java.util.List;  
import java.util.Iterator;  
import java.util.ArrayList;  
  
<parser>  
>>
```

```
ListenerFile(file, header, namedActions) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
<if(file.genPackage)>  
package <file.genPackage>;  
<endif>  
<header>  
import org.antlr.v4.runtime.tree.ParseTreeListener;
```

```

/**
 * This interface defines a complete listener for a parse tree produced by
 * { @link <file.parserName>}.
 */
public interface <file.grammarName>Listener extends ParseTreeListener {
    <file.listenerNames: {Iname |
/**
<if(file.listenerLabelRuleNames.(Iname))>
 * Enter a parse tree produced by the { @code <Iname>}
 * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>}}.
<else>
 * Enter a parse tree produced by { @link <file.parserName>#<Iname>}}.
<endif>
 * @param ctx the parse tree
 */
void enter<Iname; format="cap">(<file.parserName>.<Iname; format="cap">Context ctx);
/**
<if(file.listenerLabelRuleNames.(Iname))>
 * Exit a parse tree produced by the { @code <Iname>}
 * labeled alternative in { @link <file.parserName>#<file.listenerLabelRuleNames.(Iname)>}}.
<else>
 * Exit a parse tree produced by { @link <file.parserName>#<Iname>}}.
<endif>
 * @param ctx the parse tree
 */
void exit<Iname; format="cap">(<file.parserName>.<Iname; format="cap">Context ctx);; separator="\n">
}
>>

```

```

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>

```

```

import org.antlr.v4.runtime.ParserRuleContext;
import org.antlr.v4.runtime.tree.ErrorNode;
import org.antlr.v4.runtime.tree.TerminalNode;

```

```

/**
 * This class provides an empty implementation of { @link <file.grammarName>Listener},
 * which can be extended to create a listener which only needs to handle a subset
 * of the available methods.
 */
public class <file.grammarName>BaseListener implements <file.grammarName>Listener {
    <file.listenerNames: {Iname |

```



```

/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void enter<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { }
/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void exit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { };
separator="\n">

/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void enterEveryRule(ParserRuleContext ctx) { }
/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void exitEveryRule(ParserRuleContext ctx) { }
/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitTerminal(TerminalNode node) { }
/**
 * {@inheritDoc}
 *
 * \<p>The default implementation does nothing.\</p>
 */
@Override public void visitErrorNode(ErrorNode node) { }
}
>>

```

```

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.ParseTreeVisitor;

```

```

/**
 * This interface defines a complete generic visitor for a parse tree produced
 * by { @link <file.parserName> }.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void } for
 * operations with no return type.
 */
public interface <file.grammarName>Visitor\<T> extends ParseTreeVisitor\<T> {
    <file.visitorNames:{ lname |
/**
<if(file.visitorLabelRuleNames.(lname))>
 * Visit a parse tree produced by the { @code <lname>\}
 * labeled alternative in { @link <file.parserName>#<file.visitorLabelRuleNames.(lname)>\}.
<else>
 * Visit a parse tree produced by { @link <file.parserName>#<lname>\}.
<endif>
 * @param ctx the parse tree
 * @return the visitor result
 */
    T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx);}; separator="\n">
    }
>>

```

```

BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
package <file.genPackage>;
<endif>
<header>
import org.antlr.v4.runtime.tree.AbstractParseTreeVisitor;

```

```

/**
 * This class provides an empty implementation of { @link <file.grammarName>Visitor },
 * which can be extended to create a visitor which only needs to handle a subset
 * of the available methods.
 *
 * @param \<T> The return type of the visit operation. Use { @link Void } for
 * operations with no return type.
 */
public class <file.grammarName>BaseVisitor\<T> extends AbstractParseTreeVisitor\<T> implements
<file.grammarName>Visitor\<T> {
    <file.visitorNames:{ lname |
/**
 * { @inheritDoc\}
 *
 * \<p>The default implementation returns the result of calling
 * { @link #visitChildren\} on { @code ctx\}.\</p>

```

```

*/
@Override public T visit<lname; format="cap">(<file.parserName>.<lname; format="cap">Context ctx) { return
visitChildren(ctx); \} }; separator="\n">
}
>>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
// Generated from <grammarFileName; format="java-escape"> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <parser.name> extends <superClass; null="Parser"> {
static { RuntimeMetaData.checkVersion("<file.ANTLRVersion>", RuntimeMetaData.VERSION); }

protected static final DFA[] _decisionToDFA;
protected static final PredictionContextCache _sharedContextCache =
new PredictionContextCache();
<if(parser.tokens)>
public static final int
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
<endif>
public static final int
<parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
private static String[] makeRuleNames() {
return new String[] {
<parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};
}
public static final String[] ruleNames = makeRuleNames();

<vocabulary(parser.literalNames, parser.symbolicNames)>

@Override
public String getGrammarFileName() { return "<parser.grammarFileName; format="java-escape">"; }

@Override
public String[] getRuleNames() { return ruleNames; }

@Override
public String getSerializedATN() { return _serializedATN; }

@Override
public ATN getATN() { return _ATN; }

```

```

<namedActions.members>
<parser:(ctor)()>
<funcs; separator="\n">

<if(sempredFuncs)>
public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <parser.sempredFuncs.values:{f}
    case <f.ruleIndex>:
        return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
    }
    return true;
    }
    <sempredFuncs.values; separator="\n">
<endif>

<atn>
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
private static String[] makeLiteralNames() {
    return new String[] {
        <literalNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
    };
}
private static final String[] _LITERAL_NAMES = makeLiteralNames();
private static String[] makeSymbolicNames() {
    return new String[] {
        <symbolicNames:{t | <t>}; null="null", separator=", ", wrap, anchor>
    };
}
private static final String[] _SYMBOLIC_NAMES = makeSymbolicNames();
public static final Vocabulary VOCABULARY = new VocabularyImpl(_LITERAL_NAMES,
    _SYMBOLIC_NAMES);

/**
 * @deprecated Use {@link #VOCABULARY} instead.
 */
@Deprecated
public static final String[] tokenNames;
static {
    tokenNames = new String[_SYMBOLIC_NAMES.length];
    for (int i = 0; i < tokenNames.length; i++) {
        tokenNames[i] = VOCABULARY.getLiteralName(i);
        if (tokenNames[i] == null) {
            tokenNames[i] = VOCABULARY.getSymbolicName(i);

```

```

    }

    if (tokenNames[i] == null) {
        tokenNames[i] = "\\<INVALID>";
    }
}

@Override
@Deprecated
public String[] getTokenNames() {
    return tokenNames;
}

@Override

public Vocabulary getVocabulary() {
    return VOCABULARY;
}
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
@Override
public void action(RuleContext _localctx, int ruleIndex, int actionIndex) {
    switch (ruleIndex) {
        <recog.actionFuncs.values:{f}
        case <f.ruleIndex>:
            <f.name>_action((<f.ctxType>)_localctx, actionIndex);
            break;}; separator="\n">
    }
}
<actionFuncs.values; separator="\n">
<endif>
<if(sempredFuncs)>
@Override
public boolean sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <recog.sempredFuncs.values:{f}
        case <f.ruleIndex>:
            return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
    }
    return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

```

```

parser_ctor(p) ::= <<
public <p.name>(TokenStream input) {
    super(input);
    _interp = new ParserATNSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
}
>>

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
    switch (actionIndex) {
    <actions:{index|
case <index>:
    <actions.(index)>
    break;}; separator="\n">
    }
}
>>

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private boolean <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
    switch (predIndex) {
    <actions:{index|
case <index>:
    return <actions.(index)>;}; separator="\n">
    }
    return true;
}
>>

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name><(<args; separator=",">) throws RecognitionException {
    <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, getState())<currentRule.args:{a | ,
<a.name>}>>;
    enterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
    <namedActions.init>
    <locals; separator="\n">

```

```

try {
<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
  <code>
  <postamble; separator="\n">
  <namedActions.after>
  }
  <if(exceptions)>
  <exceptions; separator="\n">
  <else>
  catch (RecognitionException re) {
    _localctx.exception = re;
    _errHandler.reportError(this, re);
    _errHandler.recover(this, re);
  }
  <endif>
  finally {
    <finallyAction>
    exitRule();
  }
  return _localctx;
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
  namedActions,finallyAction,postamble) ::=
<<

```

```

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public final <endif><currentRule.ctxType>
<currentRule.name><(args; separator=", "> throws RecognitionException {
  return <currentRule.name>(0<currentRule.args:{a | , <a.name>}>);
}

```

```

private <currentRule.ctxType> <currentRule.name>(int _p<args:{a | , <a>}>) throws RecognitionException {
  ParserRuleContext _parentctx = _ctx;
  int _parentState = getState();
  <currentRule.ctxType> _localctx = new <currentRule.ctxType>(_ctx, _parentState<currentRule.args:{a | ,
  <a.name>}>);
  <currentRule.ctxType> _prevctx = _localctx;
  int _startState = <currentRule.startState>;
  enterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
  <namedActions.init>
  <locals; separator="\n">
  try {

```

```

<if(currentRule.hasLookaheadBlock)>
  int _alt;
<endif>
<code>
<postamble; separator="\n">
<namedActions.after>
}
catch (RecognitionException re) {
  _localctx.exception = re;
  _errHandler.reportError(this, re);
  _errHandler.recover(this, re);
}
finally {
  <finallyAction>
  unrollRecursionContexts(_parentctx);
}
return _localctx;
}
>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
enterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>
<preamble; separator="\n">
switch (_input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
<alt>
break;}; separator="\n">
default:
<error>
}
>>

```



```

LL1OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
switch (_input.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
<alt>
break;}; separator="\n">
default:
break;
}
>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<preamble; separator="\n">
if (<expr>) {
<alts; separator="\n">
}
<!else if ( !(<followExpr>) ) <error!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<preamble; separator="\n">
while (<loopExpr>) {
<alts; separator="\n">
setState(<choice.loopBackStateNumber>);
_errHandler.sync(this);
<iteration>
}
>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
setState(<choice.blockStartStateNumber>); <! alt block decision !>
_errHandler.sync(this);
<preamble; separator="\n">
do {
<alts; separator="\n">
setState(<choice.stateNumber>); <! loopback/exit decision !>
_errHandler.sync(this);
<iteration>
} while ( <loopExpr> );
>>

```

```

// LL(*) stuff

```

```

AltBlock(choice, preamble, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
<if(choice.label)><labelref(choice.label)> = _input.LT(1);<endif>
<preamble; separator="\n">
switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {
<alts:{alt |
case <i>:
<alt>
break;}; separator="\n">
}
>>

```

```

OptionalBlock(choice, alts, error) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
switch ( getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx) ) {
<alts:{alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
}
>>

```

```

StarBlock(choice, alts, sync, iteration) ::= <<
setState(<choice.stateNumber>);
_errHandler.sync(this);
_alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER ) {
if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
<iteration>
<alts> <! should only be one !>
}
setState(<choice.loopBackStateNumber>);
_errHandler.sync(this);
_alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
}
>>

```

```

PlusBlock(choice, alts, error) ::= <<
setState(<choice.blockStartStateNumber>); <! alt block decision !>
_errHandler.sync(this);
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
switch (_alt) {
<alts:{alt|
case <i><if(!choice.ast.greedy)>+1<endif>:

```

```

<alt>
break;}; separator="\n">
default:
  <error>
  }
  setState(<choice.loopBackStateNumber>); <! loopback/exit decision !>
  _errHandler.sync(this);
  _alt = getInterpreter().adaptivePredict(_input,<choice.decision>,_ctx);
  } while ( _alt!=<choice.exitAlt> && _alt!=org.antlr.v4.runtime.atn.ATN.INVALID_ALT_NUMBER );
  >>

Sync(s) ::= "sync(<s.expecting.name>);"

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
  >>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
  >>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ttype | (1L \<< <offsetShift(ttype, bits.shift)>)}; separator=" | ">)) != 0)
  %>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
  %>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<ttype>}; separator=" || ">
  %>

cases(ttypes) ::= <<
<ttypes:{t | case <t>}; separator="\n">
  >>

```

```

InvokeRule(r, argExprsChunks) ::= <<
  setState(<r.stateNumber>);
  <if(r.labels)><r.labels:{1 | <labelref(l)> =
  }><endif><r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)>,<endif><endif><argExprsChunks>
  );
>>

```

```

MatchToken(m) ::= <<
  setState(<m.stateNumber>);
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>match(<m.name>);
>>

```

```

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

```

```

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

```

```

CommonSetStuff(m, expr, capture, invert) ::= <<
  setState(<m.stateNumber>);
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }>_input.LT(1);<endif>
  <capture>
  if ( <if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr> ) ) {
    <if(m.labels)><m.labels:{1 | <labelref(l)> = (Token)}><endif>_errHandler.recoverInline(this);
  }
  else {
    if ( _input.LA(1)==Token.EOF ) matchedEOF = true;
    _errHandler.reportMatch(this);
    consume();
  }
>>

```

```

Wildcard(w) ::= <<
  setState(<w.stateNumber>);
  <if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>matchWildcard();
>>

```

```

// ACTION STUFF

```

```

Action(a, foo, chunks) ::= "<chunks>"

```

```

ArgAction(a, chunks) ::= "<chunks>"

```

```

SemPred(p, chunks, failChunks) ::= <<
  setState(<p.stateNumber>);
  if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,
  <failChunks><elseif(p.msg)>, <p.msg><endif>);
>>

```

```

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
  <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip();"
LexerMoreCommand() ::= "more();"
LexerPopModeCommand() ::= "popMode();"

LexerTypeCommand(arg, grammar) ::= "_type = <arg>";
LexerChannelCommand(arg, grammar) ::= "_channel = <arg>";
LexerModeCommand(arg, grammar) ::= "_mode = <arg>";
LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>);"

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<a.name>"
LocalRef(a) ::= "_localctx.<a.name>"
RetValRef(a) ::= "_localctx.<a.name>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<t.name>"
LabelRef(t) ::= "<ctx(t)>.<t.name>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>";

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"
InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getText():null)"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getType():0)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getLine():0)"
TokenPropertyRef_pos(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getCharPositionInLine():0)"
TokenPropertyRef_channel(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getChannel():0)"
TokenPropertyRef_index(t) ::= "<ctx(t)>.<t.label>!=null?<ctx(t)>.<t.label>.getTokenIndex():0)"
TokenPropertyRef_int(t) ::= "<ctx(t)>.<t.label>!=null?Integer.valueOf(<ctx(t)>.<t.label>.getText()):0)"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>!=null?<ctx(r)>.<r.label>.start():null)"
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>!=null?<ctx(r)>.<r.label>.stop():null)"
RulePropertyRef_text(r) ::=
"<ctx(r)>.<r.label>!=null?_input.getText(<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop):null)"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "this"

ThisRulePropertyRef_start(r) ::= "_localctx.start"

```

```

ThisRulePropertyRef_stop(r) ::= "_localctx.stop"
ThisRulePropertyRef_text(r) ::= "_input.getText(_localctx.start, _input.LT(-1))"
ThisRulePropertyRef_ctx(r) ::= "_localctx"
ThisRulePropertyRef_parser(r) ::= "this"

NonLocalAttrRef(s) ::= "(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name>"
SetNonLocalAttr(s, rhsChunks) ::=
"(<<s.ruleName; format=\"cap\">Context)getInvokingContext(<s.ruleIndex>)).<s.name> = <rhsChunks>);"

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.add(<labelref(a.label)>);"

TokenDecl(t) ::= "<TokenLabelType()> <t.name>"
TokenTypeDecl(t) ::= "int <t.name>);"
TokenListDecl(t) ::= "List<Token> <t.name> = new ArrayList<Token>()"
RuleContextDecl(r) ::= "<r.ctxName> <r.name>"
RuleContextListDecl(rdecl) ::= "List<<rdecl.ctxName>> <rdecl.name> = new ArrayList<<rdecl.ctxName>>()"

ContextTokenGetterDecl(t) ::=
"public TerminalNode <t.name>() { return getToken(<parser.name>.<t.name>, 0); }"
ContextTokenListGetterDecl(t) ::=
"public List<TerminalNode> <t.name>() { return getTokens(<parser.name>.<t.name>); }"
ContextTokenListIndexedGetterDecl(t) ::= <<
public TerminalNode <t.name>(int i) {
return getToken(<parser.name>.<t.name>, i);
}
>>
ContextRuleGetterDecl(r) ::= <<
public <r.ctxName> <r.name>() {
return getRuleContext(<r.ctxName>.class,0);
}
>>
ContextRuleListGetterDecl(r) ::= <<
public List<<r.ctxName>> <r.name>() {
return getRuleContexts(<r.ctxName>.class);
}
>>
ContextRuleListIndexedGetterDecl(r) ::= <<
public <r.ctxName> <r.name>(int i) {
return getRuleContext(<r.ctxName>.class,i);
}
>>

LexerRuleContext() ::= "RuleContext"

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/
RuleContextNameSuffix() ::= "Context"

```

```
ImplicitTokenLabel(tokenName) ::= "<tokenName>"
```

```
ImplicitRuleLabel(ruleName) ::= "<ruleName>"
```

```
ImplicitSetLabel(id) ::= "_tset<id>"
```

```
ListLabelName(label) ::= "<label>"
```

```
CaptureNextToken(d) ::= "<d.varName> = _input.LT(1);"
```

```
CaptureNextTokenType(d) ::= "<d.varName> = _input.LA(1);"
```

```
StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers)
```

```
 ::= <<
```

```
public static class <struct.name> extends
```

```
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)> implements
```

```
<interfaces; separator=", "><endif> {
```

```
<attrs:{ a | public <a>; }; separator="\n">
```

```
<getters:{ g | <g>; }; separator="\n">
```

```
<if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) { super(parent, invokingState);
```

```
}<endif>
```

```
public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{ a | , <a>>>) {
```

```
super(parent, invokingState);
```

```
<struct.ctorAttrs:{ a | this.<a.name> = <a.name>; }; separator="\n">
```

```
}
```

```
@Override public int getRuleIndex() { return RULE_<struct.derivedFromName>; }
```

```
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
```

```
public <struct.name>() { }
```

```
public void copyFrom(<struct.name> ctx) {
```

```
super.copyFrom(ctx);
```

```
<struct.attrs:{ a | this.<a.name> = ctx.<a.name>; }; separator="\n">
```

```
}
```

```
<endif>
```

```
<dispatchMethods; separator="\n">
```

```
<extensionMembers; separator="\n">
```

```
}
```

```
>>
```

```
AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
```

```
public static class <struct.name> extends <currentRule.name; format="cap">Context {
```

```
<attrs:{ a | public <a>; }; separator="\n">
```

```
<getters:{ g | <g>; }; separator="\n">
```

```
public <struct.name>(<currentRule.name; format="cap">Context ctx) { copyFrom(ctx); }
```

```
<dispatchMethods; separator="\n">
```

```
}
```

```
>>
```

```
ListenerDispatchMethod(method) ::= <<
```

```
@Override
```

```
public void <if(method.isEnter)>enter<else>exit<endif>Rule(ParseTreeListener listener) {
```

```
if ( listener instanceof <parser.grammarName>Listener )
```

```

((<parser.grammarName>Listener)listener).<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName;
format="cap">(this);
}
>>

```

```

VisitorDispatchMethod(method) ::= <<
@Override
public <T> T accept(ParseTreeVisitor<? extends T> visitor) {
if ( visitor instanceof <parser.grammarName>Visitor ) return ((<parser.grammarName>Visitor<? extends
T>)visitor).visit<struct.derivedFromName; format="cap">(this);
else return visitor.visitChildren(this);
}
>>

```

```

AttributeDecl(d) ::= "<d.type> <d.name><if(d.initValue)> = <d.initValue><endif>"

```

```

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>((<x.ctx.name>)_localctx).<endif><x.name>"

```

```

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "((<actionChunk.ctx.name>)_localctx)"

```

```

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "precpred(_ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>";
recRuleSetStopToken() ::= "_ctx.stop = _input.LT(-1);"

```

```

recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
_localctx = new <ctxName>Context(_parentctx, _parentState);
<if(label)>
<if(isListLabel)>
_localctx.<label>.add(_prevctx);
<else>
_localctx.<label> = _prevctx;
<endif>
<endif>
<if(label)>_localctx.<label> = _prevctx;<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

```

```

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
_localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
<if(label)>
<if(isListLabel)>
((<currentAltLabel; format="cap">Context)_localctx).<label>.add(_prevctx);
<else>
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;

```



```

<endif>
<endif>
pushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
>>

recRuleReplaceContext(ctxName) ::= <<
_localctx = new <ctxName>Context(_localctx);
_ctx = _localctx;
_prevctx = _localctx;
>>

recRuleSetPrevCtx() ::= <<
if ( _parseListeners!=null ) triggerExitRuleEvent();
_prevctx = _localctx;
>>

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
<if(lexerFile.genPackage)>
package <lexerFile.genPackage>;
<endif>
<namedActions.header>
import org.antlr.v4.runtime.Lexer;
import org.antlr.v4.runtime.CharStream;
import org.antlr.v4.runtime.Token;
import org.antlr.v4.runtime.TokenStream;
import org.antlr.v4.runtime.*;
import org.antlr.v4.runtime.atn.*;
import org.antlr.v4.runtime.dfa.DFA;
import org.antlr.v4.runtime.misc.*;

<lexer>
>>

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
@SuppressWarnings({"all", "warnings", "unchecked", "unused", "cast"})
public class <lexer.name> extends <superClass; null="Lexer"> {
static { RuntimeMetaData.checkVersion("<lexerFile.ANTLRVersion>", RuntimeMetaData.VERSION); }

protected static final DFA[] _decisionToDFA;
protected static final PredictionContextCache _sharedContextCache =
new PredictionContextCache();
<if(lexer.tokens)>
public static final int
<lexer.tokens:{k | <k>=<lexer.tokens.(k)>} ; separator=", ", wrap, anchor>;
<endif>
<if(lexer.channels)>

```

```

public static final int
<lexer.channels:{c | <c>=<lexer.channels.(c)>}; separator=", ", wrap, anchor>
<endif>
<if(rest(lexer.modes))>
public static final int
<rest(lexer.modes){m | <m>=<i>}; separator=", ", wrap, anchor>
<endif>
public static String[] channelNames = {
"DEFAULT_TOKEN_CHANNEL", "HIDDEN"<if (lexer.channels)>, <lexer.channels:{c| "<c>"}; separator=", ",
wrap, anchor><endif>
};

public static String[] modeNames = {
<lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor>
};

private static String[] makeRuleNames() {
return new String[] {
<lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};
}
public static final String[] ruleNames = makeRuleNames();

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

<namedActions.members>

public <lexer.name>(CharStream input) {
super(input);
_interp = new LexerATNSimulator(this, _ATN, _decisionToDFA, _sharedContextCache);
}

@Override
public String getGrammarFileName() { return "<lexer.grammarFileName>"; }

@Override
public String[] getRuleNames() { return ruleNames; }

@Override
public String getSerializedATN() { return _serializedATN; }

@Override
public String[] getChannelNames() { return channelNames; }

@Override
public String[] getModeNames() { return modeNames; }

@Override

```

```

public ATN getATN() { return _ATN; }

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>
}
>>

SerializedATN(model) ::= <<
<if(rest(model.segments))>
<! requires segmented representation !>
private static final int _serializedATNSegments = <length(model.segments)>;
<model.segments:{segment|private static final String _serializedATNSegment<i0> =
"<segment; wrap={ "+<\n><\t>" }>"; separator="\n">
public static final String _serializedATN = Utils.join(
new String[] {
<model.segments:{segment | _serializedATNSegment<i0>}; separator=",\n">
},
""
);
<else>
<! only one segment, can be inlined !>
public static final String _serializedATN =
"<model.serialized; wrap={ "+<\n><\t>" }>";
<endif>
public static final ATN _ATN =
new ATNDeserializer().deserialize(_serializedATN.toCharArray());
static {
_decisionToDFA = new DFA[_ATN.getNumberOfDecisions()];
for (int i = 0; i < _ATN.getNumberOfDecisions(); i++) {
_decisionToDFA[i] = new DFA(_ATN.getDecisionState(i), i);
}
<! org.antlr.v4.tool.DOTGenerator dot = new org.antlr.v4.tool.DOTGenerator(null);!>
<! System.out.println(dot.getDOT(_ATN.decisionToState.get(0), ruleNames, false));!>
<! System.out.println(dot.getDOT(_ATN.ruleToStartState[2], ruleNames, false));!>
}
>>

/** Using a type to init value map, try to init a type; if not in table
* must be an object, default value is "null".
*/
initValue(typeName) ::= <<
<javaTypeInitMap.(typeName)>
>>

codeFileExtension() ::= ".java"

```

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/tool/templates/codegen/Java/Java.stg

No license file was found, but licenses were detected in source scan.

/*

* Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.

* Use of this file is governed by the BSD 3-clause license that

* can be found in the LICENSE.txt file in the project root.

*/

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/RuleRefAST.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LL1AltBlock.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/JFileChooserConfirmOverwrite.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LL1Loop.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/NotAST.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/SemPred.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/GrammarAST.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ThrowEarlyExitException.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ToolANTLRParser.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/CaptureNextTokenType.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ElementFrequenciesVisitor.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LeftRecursiveRuleFunction.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/RangeAST.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_start.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/LeftRecursiveRule.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/OptionalBlock.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/DispatchMethod.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_text.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/StructDecl.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/codegen/model/Recognizer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_start.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/BlankOutputModelFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/VisitorDispatchMethod.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/GrammarInterpreterRuleContext.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/LL1OptionalBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/gui/TreeTextProvider.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/ThrowRecognitionException.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_pos.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/ast/QuantifierAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/MatchSet.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/VisitorFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/MatchNotSet.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/semantics/BlankActionSplitterListener.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/semantics/SymbolCollector.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/parse/GrammarASTAdaptor.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/analysis/LeftRecursiveRuleAltInfo.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/decl/TokenListDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/LexerFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_int.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/semantics/SemanticPipeline.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/Parser.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/target/Python3Target.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/parse/TokenVocabParser.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/tool/GrammarParserInterpreter.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/target/SwiftTarget.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/tool/DefaultToolListener.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/tool/ANTLRMessage.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/RetValRef.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/decl/ContextTokenGetterDecl.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/automata/TailEpsilonRemover.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/ThrowNoViableAlt.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/decl/Decl.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/tool/ast/OptionalBlockAST.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/parse/GrammarToken.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/gui/TreeLayoutAdaptor.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/AttributeDict.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_channel.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/Trees.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/ArgRef.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/ActionTemplate.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/LL1OptionalBlockSingleAlt.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/tool/LeftRecursionCyclesMessage.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/SrcOp.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/ActionText.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/Alternative.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/RuleElement.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/codegen/model/chunk/NonLocalAttrRef.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
 jar/org/antlr/v4/tool/AttributeResolver.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/Utils.java
 * /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/automata/LexerATNFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/TreeViewer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ErrorMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ErrorMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/CodeBlockForOuterMostAlt.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/AltBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/TestSetInline.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/PostScriptDocument.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/EscapeSequenceParsing.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_parser.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_line.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/OutputModelObject.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/CodeBlockForAlt.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/OutputModelWalker.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/Grammar.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/BasicFontMetrics.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/Graph.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/TerminalAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ErrorMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LabeledOp.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/semantics/SymbolChecks.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/ActionChunk.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/BaseVisitorFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ExceptionClause.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/BlockAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ContextGetterDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/automata/ParserATNFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ContextRuleListGetterDecl.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/DOTGenerator.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_index.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/UnicodeEscapes.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/TreePostScriptGenerator.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/FrequencySet.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/LexerGrammar.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ContextTokenListGetterDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/LexerFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Wildcard.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/target/Python2Target.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Action.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Lexer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ContextRuleListIndexedGetterDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/TestRig.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/semantics/BasicSemanticChecks.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/semantics/UseDefAnalyzer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ToolMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/target/JavaScriptTarget.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ActionSplitterListener.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/ParserFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/ActionAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LL1StarBlockSingleAlt.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/automata/ATNOptimizer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/MutableInt.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/analysis/LeftRecursionDetector.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/AltLabelStructDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/semantics/RuleCollector.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/codegen/model/chunk/QRetValueRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/dbg.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/RuleElementAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/SerializedATN.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_text.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_stop.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/ListLabelRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/analysis/LeftRecursiveRuleTransformer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/CodeGeneratorExtension.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/AltAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/automata/ATNFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/SetAttr.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/GrammarASTErrorNode.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/TokenRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ParserFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/PredAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/Target.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/DefaultOutputModelFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/semantics/AttributeChecks.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/GrammarASTVisitor.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ToolANTLRLexer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ContextTokenListIndexedGetterDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/target/CSharpTarget.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/analysis/LeftRecursiveRuleAnalyzer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_ctx.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LL1PlusBlockSingleAlt.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/automata/ATNPrinter.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_text.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Loop.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/ast/GrammarRootAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/decl/CodeBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/misc/CharSupport.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/automata/ATNVisitor.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/ast/PlusBlockAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/OutputModelFactory.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/GrammarTransformPipeline.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/ListenerFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/Attribute.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/Tool.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/decl/ContextRuleGetterDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/RuleFunction.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/CodeGenerator.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/semantics/ActionSniffer.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/decl/RuleContextDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/BaseListenerFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/target/GoTarget.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/RuleAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_parser.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/GrammarSyntaxMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/model/MatchToken.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/ActionTranslator.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/misc/OrderedHashMap.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/StarBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Sync.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/analysis/AnalysisPipeline.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ListenerDispatchMethod.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/SetNonLocalAttr.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/RulePropertyRef_ctx.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ScopeParser.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/LabelRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ModelElement.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/StarBlockAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/v4ParserException.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/LL1Choice.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/ElementListDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/LocalRef.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/AddToLabelList.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/OutputFile.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/TokenDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/target/CppTarget.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/unicode/UnicodeDataTemplateController.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/GrammarASTWithOptions.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/TokenTypeDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/BuildDependencyGenerator.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/OutputModelController.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ANTLRToolListener.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/CodeGenPipeline.java

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/LabelType.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/RuleContextListDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/ast/SetAST.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/LabelElementPair.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/decl/AttributeDecl.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/InvokeRule.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/v3TreeGrammarException.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/target/JavaTarget.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/Rule.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/Choice.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/TokenPropertyRef_type.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/PlusBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/SystemFontMetrics.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/ArgAction.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/GrammarSemanticsMessage.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/RuleSempredFunction.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/RuleActionFunction.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/CaptureNextToken.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ResyncToEndOfRuleBlock.java
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/codegen/model/chunk/ThisRulePropertyRef_stop.java
No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```
// File   : A3Lexer.g
// Author  : Jim Idle (jimi@temporal-wave.com)
// Copyright : Free BSD - See @header clause below
// Version  : First implemented as part of ANTLR 3.2 this is the self
//           hosting ANTLR 3 Lexer.
//
// Description
// -----
// This is the definitive lexer grammar for parsing ANTLR V3.x.x grammars. All other
// grammars are derived from this grammar via source code control integration (perforce)
// or by the gdiff tool.
//
// This grammar and its associated grmmars A3Parser.g and A3Walker.g exhibit the following
// traits, which are recommended for all production quality grammars:
//
// 1) They are separate grammars, not composite grammars;
// 2) They implement all supporting methods in a superclass (at least this is recommended
//    for language targets that support inheritance;
// 3) All errors are pushed as far down the parsing chain as possible, which means
//    that the lexer tries to defer error reporting to the parser, and the parser
//    tries to defer error reporting to a semantic phase consisting of a single
//    walk of the AST. The reason for this is that the error messages produced
//    from later phases of the parse will generally have better context and so
//    be more useful to the end user. Consider the message: "Syntax error at 'options'"
//    vs: "You cannot specify two options{ } sections in a single grammar file".
// 4) The lexer is 'programmed' to catch common mistakes such as unterminated literals
//    and report them specifically and not just issue confusing lexer mismatch errors.
```

```

//

/** Read in an ANTLR grammar and build an AST. Try not to do
 * any actions, just build the tree.
 *
 * The phases are:
 *
 * A3Lexer.g (this file)
 *     A3Parser.g
 *     A3Verify.g (derived from A3Walker.g)
 * assign.types.g
 * define.g
 * buildnfa.g
 * antlr.print.g (optional)
 * codegen.g
 *
 * Terence Parr
 * University of San Francisco
 * 2005
 * Jim Idle (this v3 grammar)
 * Temporal Wave LLC
 * 2009
 */
lexer grammar ANTLRLexer;

// =====
// Note that while this grammar does not care about order of constructs
// that don't really matter, such as options before @header etc, it must first
// be parsed by the original v2 parser, before it replaces it. That parser does
// care about order of structures. Hence we are constrained by the v2 parser
// for at least the first bootstrap release that causes this parser to replace
// the v2 version.
// =====

// -----
// Options
//
// V3 option directives to tell the tool what we are asking of it for this
// grammar.
//
options {

// Target language is Java, which is the default but being specific
// here as this grammar is also meant as a good example grammar for
// for users.
//
language = Java;

```

```

// The super class that this lexer should expect to inherit from, and
// which contains any and all support routines for the lexer. This is
// commented out in this baseline (definitive or normative grammar)
// - see the ANTLR tool implementation for hints on how to use the super
// class
//
//superclass = AbstractA3Lexer;
}

```

```
tokens { SEMPRED; TOKEN_REF; RULE_REF; LEXER_CHAR_SET; ARG_ACTION; }
```

```

// Include the copyright in this source and also the generated source
//

```

```

@lexer::header {
/*
[The "BSD licence"]
Copyright (c) 2005-2009 Terence Parr
All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```

*/
package org.antlr.v4.parse;
import org.antlr.v4.tool.*;
import org.antlr.v4.runtime.misc.Interval;
}

```

```
@members {
```

```

public static final int COMMENTS_CHANNEL = 2;

public CommonTokenStream tokens; // track stream we push to; need for context info
public boolean isLexerRule = false;

public void grammarError(ErrorType etype, org.antlr.runtime.Token token, Object... args) { }

/** scan backwards from current point in this.tokens list
 * looking for the start of the rule or subrule.
 * Return token or null if for some reason we can't find the start.
 */
public Token getRuleOrSubruleStartToken() {
    if ( tokens==null ) return null;
    int i = tokens.index();
    int n = tokens.size();
    if ( i>=n ) i = n-1; // seems index == n as we lex
    while ( i>=0 && i<n ) {
        int ttype = tokens.get(i).getType();
        if ( ttype == LPAREN || ttype == TOKEN_REF || ttype == RULE_REF ) {
            return tokens.get(i);
        }
        i--;
    }
    return null;
}

// -----
// Comments
//
// ANTLR comments can be multi or single line and we don't care
// which particularly. However we also accept Javadoc style comments
// of the form: /** ... */ and we do take care to distinguish those
// from ordinary multi-line comments
// Note how we guide the lexical PATH because we want to issue a decriptive
// error message in case of a standalone '/' character, which makes no
// sense in ANTLR source code. We also trap unterminated multi-line comments
//
fragment DOC_COMMENT : ;
COMMENT
@init {

// Record the start line and offsets as if we need to report an
// unterminated comment, then we want to show the start of the comment
// we think is broken, not the end, where people will have to try and work
// it out themselves.
//
int startLine = $line;

```



```

int offset = getCharPositionInLine();
}
: // Eat the first character only, then see if we have a comment
  // or something silly.
  //
  '/' // Comment introducer

(
  // Single line comment, possibly with embedded src/line directives
  // in a similar style to the C pre-processor, allowing generated
  // code to refer the programmer back to the original source code
  // in case of error.
  //
  '/'
  (
    (' $ANTLR')=> ' $ANTLR' SRC
    | ~(NLCHARS)*
  )

| // Multi-line comment, which may be a documentation comment
  // if it starts /** (note that we protect against accidentally
  // recognizing a comment /**/ as a documentation comment
  //
  /** (
    { input.LA(2) != '/' }?=> '*' { $type = DOC_COMMENT; }
    | { true }?=> // Required to cover all alts with predicates
  )

  // Should we support embedded multiline comments here?
  //
  (
    // Pick out end of multiline comment and exit the loop
    // if we find it.
    //
    { !(input.LA(1) == '*' && input.LA(2) == '/') }?

    // Anything else other than the non-greedy match of
    // the comment close sequence
    //
    .
  )*
  (
    // Look for the comment terminator, but if it is accidentally
    // unterminated, then we will hit EOF, which will trigger the
    // epsilon alt and hence we can issue an error message relative
    // to the start of the unterminated multi-line comment
    //
    /**/

```

```

        |// Unterminated comment!
        //
        {
            // ErrorManager.msg(Msg.UNTERMINATED_DOC_COMMENT, startLine, offset, $pos, startLine,
offset, $pos, (Object)null);
        }
    )

    |// There was nothing that made sense following the opening '/' and so
    // we issue an error regarding the malformed comment
    //
    {
        // TODO: Insert error message relative to comment start
        //
    }
)
{
    // We do not wish to pass the comments in to the parser. If you are
    // writing a formatter then you will want to preserve the comments off
    // channel, but could just skip and save token space if not.
    //
    $channel=COMMENTS_CHANNEL;
}
;

```

ARG_OR_CHARSET

```

options {k=1;}
: {!isLexerRule}?=> LEXER_CHAR_SET {$type=LEXER_CHAR_SET;}
| {!isLexerRule}?=> ARG_ACTION
{
    $type=ARG_ACTION;
    // Set the token text to our gathered string minus outer [ ]
    String t = $text;
    t = t.substring(1,t.length()-1);
    setText(t);
}
;

```

fragment

LEXER_CHAR_SET

```

: '['
( '\\ ~(\'r'\n')
| ~(\'r'\n'\\')
)*
']'
;

```

```

// -----
// Argument specs
//
// Certain argument lists, such as those specifying call parameters
// to a rule invocation, or input parameters to a rule specification
// are contained within square brackets. In the lexer we consume them
// all at once and sort them out later in the grammar analysis.
//
fragment
ARG_ACTION
: '['
  (
    ARG_ACTION

    | ("")=>ACTION_STRING_LITERAL

    | ("\")=>ACTION_CHAR_LITERAL

    | ~('[\'])
  )*
  '['
;

// -----
// Actions
//
// Other than making sure to distinguish between { and } embedded
// within what we have assumed to be literals in the action code, the
// job of the lexer is merely to gather the code within the action
// (delimited by {}) and pass it to the parser as a single token.
// We know that this token will be asked for its text somewhere
// in the upcoming parse, so setting the text here to exclude
// the delimiting {} is no additional overhead.
//
ACTION
: NESTED_ACTION
( '?' { $type = SEMPRED; }
( (WSNLCHARS* '=>') => WSNLCHARS* '=>' // v3 gated sempred
{
Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
t.setLine(state.tokenStartLine);
t.setText(state.text);
t.setCharPositionInLine(state.tokenStartCharPositionInLine);
grammarError(ErrorType.V3_GATED_SEMPRED, t);
}
)?
)?

```

```

;

// -----
// Action structure
//
// Many language targets use { } as block delimiters and so we
// must recursively match { } delimited blocks to balance the
// braces. Additionally, we must make some assumptions about
// literal string representation in the target language. We assume
// that they are delimited by ' or " and so consume these
// in their own alts so as not to inadvertently match { }.
// This rule calls itself on matching a {
//
fragment
NESTED_ACTION
@init {

// Record the start line and offsets as if we need to report an
// unterminated block, then we want to show the start of the comment
// we think is broken, not the end, where people will have to try and work
// it out themselves.
//
int startLine = getLine();
int offset = getCharPositionInLine();
}

: // Action and other blocks start with opening {
//
'{'
(
// And now we can match one of a number of embedded
// elements within the action until we find a
// } that balances the opening {. If we do not find
// the balanced } then we will hit EOF and can issue
// an error message about the brace that we believe to
// be mismatched. This won't be foolproof but we will
// be able to at least report an error against the
// opening brace that we feel is in error and this will
// guide the user to the correction as best we can.
//
// An embedded {} block
//
NESTED_ACTION

| // What appears to be a literal
//

```

```

ACTION_CHAR_LITERAL

|// We have assumed that the target language has C/Java
// type comments.
//
COMMENT

|// What appears to be a literal
//
ACTION_STRING_LITERAL

|// What appears to be an escape sequence
//
ACTION_ESC

|// Some other single character that is not
// handled above
//
~(\\|\"|\'|{|})

)*

(
// Correctly balanced closing brace
//
}'

|// Looks like have an imbalanced {} block, report
// with respect to the opening brace.
//
{
// TODO: Report imbalanced {}
System.out.println("Block starting at line " + startLine + " offset " + (offset+1) + " contains imbalanced {} or
is missing a }");
}
)
;

// Keywords
// -----
// keywords used to specify ANTLR v3 grammars. Keywords may not be used as
// labels for rules or in any other context where they would be ambiguous
// with the keyword vs some other identifier
// OPTIONS, TOKENS, and CHANNELS must also consume the opening brace that captures
// their option block, as this is the easiest way to parse it separate
// to an ACTION block, despite it using the same {} delimiters.
//

```

```

OPTIONS    : 'options' WSNLCHARS* '{' ;
TOKENS_SPEC : 'tokens' WSNLCHARS* '{' ;
CHANNELS   : 'channels' WSNLCHARS* '{' ;

IMPORT     : 'import'          ;
FRAGMENT   : 'fragment'       ;
LEXER      : 'lexer'          ;
PARSER     : 'parser'         ;
GRAMMAR    : 'grammar'        ;
TREE_GRAMMAR : 'tree' WSNLCHARS* 'grammar' ;
PROTECTED  : 'protected'      ;
PUBLIC     : 'public'         ;
PRIVATE    : 'private'        ;
RETURNS    : 'returns'        ;
LOCALS     : 'locals'         ;
THROWS     : 'throws'         ;
CATCH      : 'catch'          ;
FINALLY    : 'finally'        ;
MODE       : 'mode'           ;

// -----
// Punctuation
//
// Character sequences used as separators, delimiters, operators, etc
//
COLON      : ':'
    {
        // scan backwards, looking for a RULE_REF or TOKEN_REF.
        // which would indicate the start of a rule definition.
        // If we see a LPAREN, then it's the start of the subrule.
        // this.tokens is the token string we are pushing into, so
        // just loop backwards looking for a rule definition. Then
        // we set isLexerRule.
        Token t = getRuleOrSubruleStartToken();
        if ( t!=null ) {
            if ( t.getType()==RULE_REF ) isLexerRule = false;
            else if ( t.getType()==TOKEN_REF ) isLexerRule = true;
            // else must be subrule; don't alter context
        }
    }
;

COLONCOLON : '::'          ;
COMMA      : ','          ;
SEMI       : ';'          ;
LPAREN     : '('          ;
RPAREN     : ')'          ;
RARROW     : '->'         ;
LT         : '<'          ;

```

```

GT      : '>'      ;
ASSIGN  : '='      ;
QUESTION : '?'      ;
SYNPRED : '=>'    ;
    {
        Token t = new CommonToken(input, state.type, state.channel,
            state.tokenStartCharIndex, getCharIndex()-1);
        t.setLine(state.tokenStartLine);
        t.setText(state.text);
        t.setCharPositionInLine(state.tokenStartCharPositionInLine);
        grammarError(ErrorType.V3_SYNPRED, t);
        $channel=HIDDEN;
    }
    ;
STAR    : '*'      ;
PLUS    : '+'      ;
PLUS_ASSIGN : '+='  ;
OR      : '|'      ;
DOLLAR  : '$'      ;
DOT     : '.'      ; // can be WILDCARD or DOT in qid or imported rule ref
RANGE   : '..'     ;
AT      : '@'      ;
POUND   : '#'      ;
NOT     : '~'      ;
RBRACE  : '}'      ;

```

/** Allow unicode rule/token names */

```

ID : a=NameStartChar NameChar*
    {
        if ( Grammar.isTokenName($a.text) ) $type = TOKEN_REF;
        else $type = RULE_REF;
    }
    ;

```

fragment

```

NameChar : NameStartChar
    | '0'..'9'
    | '_'
    | '\u00B7'
    | '\u0300'..'u036F'
    | '\u203F'..'u2040'
    ;

```

fragment

```

NameStartChar
    : 'A'..'Z' | 'a'..'z'
    | '\u00C0'..'u00D6'
    | '\u00D8'..'u00F6'

```

```

    | '\u00F8'..\u02FF'
    | '\u0370'..\u037D'
    | '\u037F'..\u1FFF'
    | '\u200C'..\u200D'
    | '\u2070'..\u218F'
    | '\u2C00'..\u2FEF'
    | '\u3001'..\uD7FF'
    | '\uF900'..\uFDCF'
    | '\uFDF0'..\uFEFE'
    | '\uFF00'..\uFFFD'
; // ignores | [\u10000-\uEFFFF] ;

// -----
// Literals embedded in actions
//
// Note that we have made the assumption that the language used within
// actions uses the fairly standard " and ' delimiters for literals and
// that within these literals, characters are escaped using the \ character.
// There are some languages which do not conform to this in all cases, such
// as by using /string/ and so on. We will have to deal with such cases if
// if they come up in targets.
//
//
// Within actions, or other structures that are not part of the ANTLR
// syntax, we may encounter literal characters. Within these, we do
// not want to inadvertently match things like '}' and so we eat them
// specifically. While this rule is called CHAR it allows for the fact that
// some languages may use/allow ' as the string delimiter.
//
fragment
ACTION_CHAR_LITERAL
: "\"" ((\\)=>ACTION_ESC | ~\" )* "\""
;

// Within actions, or other structures that are not part of the ANTLR
// syntax, we may encounter literal strings. Within these, we do
// not want to inadvertently match things like '}' and so we eat them
// specifically.
//
fragment
ACTION_STRING_LITERAL
: "\"" ((\\)=>ACTION_ESC | ~\" )* "\""
;

// Within literal strings and characters that are not part of the ANTLR
// syntax, we must allow for escaped character sequences so that we do not
// inadvertently recognize the end of a string or character when the terminating
// delimiter has been escaped.

```



```

//
fragment
ACTION_ESC
: '\\' .
;

// -----
// Integer
//
// Obviously (I hope) match an arbitrary long sequence of digits.
//
INT : ('0'..'9')+
;

// -----
// Source spec
//
// A fragment rule for picking up information about an originating
// file from which the grammar we are parsing has been generated. This allows
// ANTLR to report errors against the originating file and not the generated
// file.
//
fragment
SRC : 'src' WSCHARS+ file=ACTION_STRING_LITERAL WSCHARS+ line=INT
{
    // TODO: Add target specific code to change the source file name and current line number
    //
}
;

// -----
// Literal string
//
// ANTLR makes no distinction between a single character literal and a
// multi-character string. All literals are single quote delimited and
// may contain unicode escape sequences of the form \uxxxx or \u{xxxxxx},
// where x is a valid hexadecimal number.
STRING_LITERAL
: '\" ( ( ESC_SEQ | ~('\\|\"|\"r|\"n') ) ) *
( '\"
| // Unterminated string literal
{
    Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
    t.setLine(state.tokenStartLine);
    t.setText(state.text);
    t.setCharacterPositionInLine(state.tokenStartCharPositionInLine);
    grammarError(ErrorType.UNTERMINATED_STRING_LITERAL, t);
}

```

```

    )
;

// A valid hex digit specification
//
fragment
HEX_DIGIT : ('0'..'9'|'a'..'f'|'A'..'F') ;

// Any kind of escaped character that we can embed within ANTLR
// literal strings.
//
fragment
ESC_SEQ
: '\\'
(
    // The standard escaped character set such as tab, newline, etc...
    'b'|'t'|'n'|'f'|'r'|'\''|'\"'

    | // A Java style Unicode escape sequence
    UNICODE_ESC

    | // A Swift/Hack style Unicode escape sequence
    UNICODE_EXTENDED_ESC

    | // An illegal escape sequence
    ~(b|'t'|'n'|'f'|'r'|'\''|'\"|'u') // \x for any invalid x (make sure to match char here)
    {
        Token t = new CommonToken(input, state.type, state.channel, getCharIndex()-2, getCharIndex()-1);
        t.setText(t.getText());
        t.setLine(input.getLine());
        t.setCharPositionInLine(input.getCharPositionInLine()-2);
        grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t, input.substring(getCharIndex()-
2,getCharIndex()-1));
    }
)
;

fragment
UNICODE_ESC
@init {

// Flag to tell us whether we have a valid number of
// hex digits in the escape sequence
//
int hCount = 0;
}
: 'u' // Leadin for unicode escape sequence

```

```

// We now require 4 hex digits. Note though
// that we accept any number of characters
// and issue an error if we do not get 4. We cannot
// use an infinite count such as + because this
// might consume too many, so we lay out the lexical
// options and issue an error at the invalid paths.
//
(
  (
    HEX_DIGIT { hCount++; }
    (
      HEX_DIGIT { hCount++; }
      (
        HEX_DIGIT { hCount++; }
        (
          // Four valid hex digits, we are good
          //
          HEX_DIGIT { hCount++; }

          |// Three valid digits
        )

        |// Two valid digits
      )

      |// One valid digit
    )
  )
  |// No valid hex digits at all
)

// Now check the digit count and issue an error if we need to
//
{
  if (hCount < 4) {
    Interval badRange = Interval.of(getCharIndex()-2-hCount, getCharIndex());
    String lastChar = input.substring(badRange.b, badRange.b);
    if ( lastChar.codePointAt(0)=="\\" ) {
      badRange.b--;
    }
    String bad = input.substring(badRange.a, badRange.b);
    Token t = new CommonToken(input, state.type, state.channel, badRange.a, badRange.b);
    t.setLine(input.getLine());
    t.setCharPositionInLine(input.getCharPositionInLine()-hCount-2);
    grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t, bad);
  }
}
;

```

```

fragment
UNICODE_EXTENDED_ESC
: 'u{' // Leadin for unicode extended escape sequence

    HEX_DIGIT+ // One or more hexadecimal digits

}' // Leadout for unicode extended escape sequence

// Now check the digit count and issue an error if we need to
{
    int numDigits = getCharIndex()-state.tokenStartCharIndex-6;
    if (numDigits > 6) {
        Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-
1);
        t.setText(t.getText());
        t.setLine(input.getLine());
        t.setCharPositionInLine(input.getCharPositionInLine()-numDigits);
        grammarError(ErrorType.INVALID_ESCAPE_SEQUENCE, t,
input.substring(state.tokenStartCharIndex,getCharIndex()-1));
    }
}
;

// -----
// Whitespace
//
// Characters and character constructs that are of no import
// to the parser and are used to make the grammar easier to read
// for humans.
//
WS
: (
    ' '
    | '\t'
    | '\r'
    | '\n'
    | '\f'
)+
{$channel=HIDDEN;}
;

// A fragment rule for use in recognizing end of line in
// rules like COMMENT.
//
fragment
NLCHARS
: '\n' | '\r'

```

```

;

// A fragment rule for recognizing traditional whitespace
// characters within lexer rules.
//
fragment
WSCHARS
: ' ' | '\t' | '\f'
;

// A fragment rule for recognizing both traditional whitespace and
// end of line markers, when we don't care to distinguish but don't
// want any action code going on.
//
fragment
WSNLCHARS
: ' ' | '\t' | '\f' | '\n' | '\r'
;

// This rule allows ANTLR 4 to parse grammars using the UTF-8 encoding with a
// byte order mark. Since this Unicode character doesn't appear as a token
// anywhere else in the grammar, we can simply skip all instances of it without
// problem. This rule will not break usage of \uFEFF inside a LEXER_CHAR_SET or
// STRING_LITERAL.
UnicodeBOM
: '\uFEFF' {skip();}
;

// -----
// Illegal Character
//
// This is an illegal character trap which is always the last rule in the
// lexer specification. It matches a single character of any value and being
// the last rule in the file will match when no other rule knows what to do
// about the character. It is reported as an error but is not passed on to the
// parser. This means that the parser to deal with the gramamr file anyway
// but we will not try to analyse or code generate from a file with lexical
// errors.
//
ERRCHAR
: .
{
    Token t = new CommonToken(input, state.type, state.channel, state.tokenStartCharIndex, getCharIndex()-1);
    t.setLine(state.tokenStartLine);
    t.setText(state.text);
    t.setCharPositionInLine(state.tokenStartCharPositionInLine);
    String msg = getTokenErrorDisplay(t) + " came as a complete surprise to me";
    grammarError(ErrorType.SYNTAX_ERROR, t, msg);
}

```

```
state.syntaxErrors++;
skip();
}
;
```

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ANTLRLexer.g
No license file was found, but licenses were detected in source scan.

```
/*
 * [The "BSD license"]
 * Copyright (c) 2012-2016 Terence Parr
 * Copyright (c) 2012-2016 Sam Harwell
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

/** templates used to generate make-compatible dependencies */

/** Generate "f : x, y, z" dependencies for input
 * dependencies and generated files. in and out
 * are File objects. For example, you can say
 * <f.canonicalPath>
 */
dependencies(grammarFileName,in,out) ::= <<
<if(in)><grammarFileName>: <in; separator=", "><endif>
```

```
<out:{f | <f> : <grammarFileName>}; separator="\n">
```

```
>>
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/tool/templates/depend.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2010 Terence Parr
```

```
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without  
modification, are permitted provided that the following conditions  
are met:
```

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

```
THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR  
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES  
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.  
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,  
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT  
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF  
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ATNBuilder.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2012-2016 Terence Parr
```

```
* Copyright (c) 2012-2016 Sam Harwell
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without  
* modification, are permitted provided that the following conditions  
* are met:
```

```
*
```

- * 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

tree grammar ATNBuilder;

```
options {
    language = Java;
    tokenVocab = ANTLRParser;
    ASTLabelType = GrammarAST;
    // filter = true;
}
```

// Include the copyright in this source and also the generated source

@header {

/*

[The "BSD license"]

Copyright (c) 2010 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
package org.antlr.v4.parse;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import org.antlr.v4.automata.ATNFactory;
}
```

```
@members {
    ATNFactory factory;
    public ATNBuilder(TreeNodeStream input, ATNFactory factory) {
        this(input);
        this.factory = factory;
    }
}
```

dummy : block[null] ; // avoid error about no start rule

ruleBlock[GrammarAST ebnfRoot] returns [ATNFactory.Handle p]

```
@init {
    List<ATNFactory.Handle> alts = new ArrayList<ATNFactory.Handle>();
    int alt = 1;
    factory.setCurrentOuterAlt(alt);
}
: ^(BLOCK
    (^ (OPTIONS .*))?)?
    ( a=alternative
        {alts.add($a.p); factory.setCurrentOuterAlt(++alt);}
    )+
)
{ $p = factory.block((BlockAST)$BLOCK, ebnfRoot, alts); }
;
```

block[GrammarAST ebnfRoot] returns [ATNFactory.Handle p]

```
@init {List<ATNFactory.Handle> alts = new ArrayList<ATNFactory.Handle>();}
: ^(BLOCK (^ (OPTIONS .*))?)? (a=alternative {alts.add($a.p);})+
{ $p = factory.block((BlockAST)$BLOCK, ebnfRoot, alts); }
;
```

alternative returns [ATNFactory.Handle p]

```
@init {List<ATNFactory.Handle> els = new ArrayList<ATNFactory.Handle>();}
: ^(LEXER_ALT_ACTION a=alternative lexerCommands)
{ $p = factory.lexerAltCommands($a.p,$lexerCommands.p); }
| ^(ALT elementOptions? EPSILON) { $p = factory.epsilon($EPSILON); }
```

```

| ^(ALT elementOptions? (e=element {els.add($e.p);})+) {$p = factory.alt(els);}
;

lexerCommands returns [ATNFactory.Handle p]
@init {List<ATNFactory.Handle> cmds = new ArrayList<ATNFactory.Handle>();}
: (c=lexerCommand {if ($c.cmd != null) cmds.add($c.cmd);})+
{
    $p = factory.alt(cmds);
}
;

lexerCommand returns [ATNFactory.Handle cmd]
: ^(LEXER_ACTION_CALL ID lexerCommandExpr)
    {$cmd = factory.lexerCallCommand($ID, $lexerCommandExpr.start);}
| ID
    {$cmd = factory.lexerCommand($ID);}
;

lexerCommandExpr
: ID
| INT
;

element returns [ATNFactory.Handle p]
: labeledElement {$p = $labeledElement.p;}
| atom {$p = $atom.p;}
| subrule {$p = $subrule.p;}
| ACTION {$p = factory.action((ActionAST)$ACTION);}
| SEMPRED {$p = factory.sempred((PredAST)$SEMPRED);}
| ^(ACTION .) {$p = factory.action((ActionAST)$ACTION);}
| ^(SEMPRED .) {$p = factory.sempred((PredAST)$SEMPRED);}
| ^(NOT b=blockSet[true]) {$p = $b.p;}
| LEXER_CHAR_SET {$p = factory.charSetLiteral($start);}
;

astOperand returns [ATNFactory.Handle p]
: atom {$p = $atom.p;}
| ^(NOT blockSet[true]) {$p = $blockSet.p;}
;

labeledElement returns [ATNFactory.Handle p]
: ^(ASSIGN ID element) {$p = factory.label($element.p);}
| ^(PLUS_ASSIGN ID element) {$p = factory.listLabel($element.p);}
;

subrule returns [ATNFactory.Handle p]
: ^(OPTIONAL block[$start]) {$p = $block.p;}
| ^(CLOSURE block[$start]) {$p = $block.p;}

```

```

| ^(POSITIVE_CLOSURE block[$start]) {$p = $block.p;}
| block[null]    {$p = $block.p;}
;

blockSet[boolean invert] returns [ATNFactory.Handle p]
@init {List<GrammarAST> alts = new ArrayList<GrammarAST>();}
: ^(SET (setElement {alts.add($setElement.start);})+) {$p = factory.set($start, alts, $invert);}
;

/** Don't combine with atom otherwise it will build spurious ATN nodes */
setElement
: ^(STRING_LITERAL .)
| ^(TOKEN_REF .)
| STRING_LITERAL
| TOKEN_REF
| ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
| LEXER_CHAR_SET
;

atom returns [ATNFactory.Handle p]
: range    {$p = $range.p;}
| ^(DOT ID terminal) {$p = $terminal.p;}
| ^(DOT ID ruleref) {$p = $ruleref.p;}
| ^(WILDCARD .) {$p = factory.wildcard($start);}
| WILDCARD {$p = factory.wildcard($start);}
| blockSet[false] {$p = $blockSet.p;}
| terminal {$p = $terminal.p;}
| ruleref {$p = $ruleref.p;}
;

ruleref returns [ATNFactory.Handle p]
: ^(RULE_REF ARG_ACTION? ^(ELEMENT_OPTIONS .*)) {$p = factory.ruleRef($RULE_REF);}
| ^(RULE_REF ARG_ACTION?) {$p = factory.ruleRef($RULE_REF);}
| RULE_REF {$p = factory.ruleRef($RULE_REF);}
;

range returns [ATNFactory.Handle p]
: ^(RANGE a=STRING_LITERAL b=STRING_LITERAL) {$p = factory.range($a,$b);}
;

terminal returns [ATNFactory.Handle p]
: ^(STRING_LITERAL .) {$p = factory.stringLiteral((TerminalAST)$start);}
| STRING_LITERAL {$p = factory.stringLiteral((TerminalAST)$start);}
| ^(TOKEN_REF ARG_ACTION .) {$p = factory.tokenRef((TerminalAST)$start);}
| ^(TOKEN_REF .) {$p = factory.tokenRef((TerminalAST)$start);}
| TOKEN_REF {$p = factory.tokenRef((TerminalAST)$start);}
;

```

```
elementOptions
: ^(ELEMENT_OPTIONS elementOption*)
;
```

```
elementOption
: ID
| ^(ASSIGN ID ID)
| ^(ASSIGN ID STRING_LITERAL)
| ^(ASSIGN ID ACTION)
| ^(ASSIGN ID INT)
;
```

Found in path(s):

```
*/opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ATNBuilder.g
No license file was found, but licenses were detected in source scan.
```

```
/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

```
tree grammar BlockSetTransformer;
options {
```

```

language = Java;
tokenVocab = ANTLRParser;
ASTLabelType = GrammarAST;
output = AST;
filter = true;
}

@header {
package org.antlr.v4.parse;
import org.antlr.v4.misc.Utills;
import org.antlr.v4.misc.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import java.util.List;
import java.util.Set;
import java.util.HashSet;
import java.util.ArrayList;
import org.antlr.v4.runtime.misc.IntervalSet;
}

@members {
public String currentRuleName;
public GrammarAST currentAlt;
public Grammar g;
public BlockSetTransformer(TreeNodeStream input, Grammar g) {
    this(input, new RecognizerSharedState());
    this.g = g;
}
}

topdown
: ^(RULE (id=TOKEN_REF|id=RULE_REF) {currentRuleName=$id.text;} .+)
| setAlt
| ebnfBlockSet
| blockSet
;

setAlt
: {inContext("RULE BLOCK")}?
ALT {currentAlt = $start;}
;

// (BLOCK (ALT (+ (BLOCK (ALT INT) (ALT ID))))))
ebnfBlockSet
@after {
GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
: ^(ebnfSuffix blockSet) -> ^(ebnfSuffix ^(BLOCK<BlockAST> ^(ALT<AltAST> blockSet)))

```

```

;

ebnfSuffix
@after {$tree = (GrammarAST)adaptor.dupNode($start);}
: OPTIONAL
| CLOSURE
| POSITIVE_CLOSURE
;

blockSet
@init {
boolean inLexer = Grammar.isTokenName(currentRuleName);
}
@after {
GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
: {inContext("RULE")}? // top-level: rule block and > 1 alt
^(BLOCK ^(alt=ALT elementOptions? {((AltAST)$alt).altLabel==null}? setElement[inLexer]) ( ^ALT
elementOptions? setElement[inLexer]) )+)
-> ^(BLOCK<BlockAST>[$BLOCK.token] ^(ALT<AltAST>[$BLOCK.token,"ALT"] ^(SET[$BLOCK.token,
"SET"] setElement+)))
| {!inContext("RULE")}? // if not rule block and > 1 alt
^(BLOCK ^(ALT elementOptions? setElement[inLexer]) ( ^ALT elementOptions? setElement[inLexer]) )+)
-> ^(SET[$BLOCK.token, "SET"] setElement+)
;

setElement[boolean inLexer]
@after {
GrammarTransformPipeline.setGrammarPtr(g, $tree);
}
: ( ^(a=STRING_LITERAL elementOptions) {!inLexer ||
CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1}?
| a=STRING_LITERAL {!inLexer || CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1}?
| {!inLexer}?=> ^(TOKEN_REF elementOptions)
| {!inLexer}?=> TOKEN_REF
| {inLexer}?=> ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
{CharSupport.getCharValueFromGrammarCharLiteral($a.getText())!=-1 &&
CharSupport.getCharValueFromGrammarCharLiteral($b.getText())!=-1}?
)
;

elementOptions
: ^(ELEMENT_OPTIONS elementOption*)
;

elementOption
: ID
| ^(ASSIGN id=ID v=ID)

```

```
| ^ (ASSIGN ID v=STRING_LITERAL)
| ^ (ASSIGN ID v=ACTION)
| ^ (ASSIGN ID v=INT)
;
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/parse/BlockSetTransformer.g
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2016, Mike Lischke
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

```
* modification, are permitted provided that the following conditions
```

```
* are met:
```

```
*
```

```
* 1. Redistributions of source code must retain the above copyright
```

```
* notice, this list of conditions and the following disclaimer.
```

```
* 2. Redistributions in binary form must reproduce the above copyright
```

```
* notice, this list of conditions and the following disclaimer in the
```

```
* documentation and/or other materials provided with the distribution.
```

```
* 3. The name of the author may not be used to endorse or promote products
```

```
* derived from this software without specific prior written permission.
```

```
*
```

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```

```
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

```
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

```
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
```

```
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
```

```
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/codegen/Cpp/Files.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD licence"]
```

```
Copyright (c) 2005-20012 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ANTLRParser.java

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2013 Terence Parr

* Copyright (c) 2013 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

// args must be <object-model-object>, <fields-resulting-in-STs>

```
ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<namedActions.header>
using System;
using System.IO;
using System.Text;
using System.Diagnostics;
using System.Collections.Generic;
using Antlr4.Runtime;
using Antlr4.Runtime.Atn;
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using DFA = Antlr4.Runtime.Dfa.DFA;

<parser>
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>
```

```
ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using IParseTreeListener = Antlr4.Runtime.Tree.IParseTreeListener;
using IToken = Antlr4.Runtime.IToken;

/// <summary>
/// This interface defines a complete listener for a parse tree produced by
/// <see cref="<csIdentifier.(file.parserName)>"/>.

```

```

/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Listener : IParseTreeListener {
  <file.listenerNames: { Iname |
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Enter a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>"/>.
<else>
/// Enter a parse tree produced by \<see cref="<file.parserName>.<Iname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Enter<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context
context);
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Exit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>"/>.
<else>
/// Exit a parse tree produced by \<see cref="<file.parserName>.<Iname>"/>.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
void Exit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context
context);}; separator="\n"
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>

```

```

BaseListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>

```

```

using Antlr4.Runtime.Misc;
using IErrorNode = Antlr4.Runtime.Tree.IErrorNode;
using ITerminalNode = Antlr4.Runtime.Tree.ITerminalNode;
using IToken = Antlr4.Runtime.IToken;
using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

```

```

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Listener"/>,

```

```

/// which can be extended to create a listener which only needs to handle a subset
/// of the available methods.
/// \</summary>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseListener : I<file.grammarName>Listener {
    <file.listenerNames> { Iname |
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Enter a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>">/.
<else>
/// Enter a parse tree produced by \<see cref="<file.parserName>.<Iname>">/.
<endif>
/// \<para>The default implementation does nothing.\</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
public virtual void Enter<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
format="cap">Context context) { \}
/// \<summary>
<if(file.listenerLabelRuleNames.(Iname))>
/// Exit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.listenerLabelRuleNames.(Iname)>">/.
<else>
/// Exit a parse tree produced by \<see cref="<file.parserName>.<Iname>">/.
<endif>
/// \<para>The default implementation does nothing.\</para>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
public virtual void Exit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
format="cap">Context context) { \} }; separator="\n"

/// \<inheritdoc>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void EnterEveryRule([NotNull] ParserRuleContext context) { }
/// \<inheritdoc>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void ExitEveryRule([NotNull] ParserRuleContext context) { }
/// \<inheritdoc>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void VisitTerminal([NotNull] ITerminalNode node) { }
/// \<inheritdoc>
/// \<remarks>The default implementation does nothing.\</remarks>
public virtual void VisitErrorNode([NotNull] IErrorNode node) { }
}
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>

```

>>

```
VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;

/// \<summary>
/// This interface defines a complete generic visitor for a parse tree produced
/// by \<see cref="<csIdentifier.(file.parserName)>">/.
/// \</summary>
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public interface I<file.grammarName>Visitor<Result> : IParseTreeVisitor<Result> {
    <file.visitorNames>: { Iname |
/// \<summary>
<if(file.visitorLabelRuleNames.(Iname))>
/// Visit a parse tree produced by the \<c><Iname>\</c>
/// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(Iname)>">/.
<else>
/// Visit a parse tree produced by \<see cref="<file.parserName>.<Iname>">/.
<endif>
/// \</summary>
/// \<param name="context">The parse tree.\</param>
/// \<return>The visitor result.\</return>
Result Visit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname; format="cap">Context
context);}; separator="\n"
    }
<if(file.genPackage)>
} // namespace <file.genPackage>
<endif>
>>
```

```
BaseVisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
<if(file.genPackage)>
namespace <file.genPackage> {
<endif>
<header>
using Antlr4.Runtime.Misc;
using Antlr4.Runtime.Tree;
using IToken = Antlr4.Runtime.IToken;
```

```

using ParserRuleContext = Antlr4.Runtime.ParserRuleContext;

/// \<summary>
/// This class provides an empty implementation of \<see cref="I<file.grammarName>Visitor{Result}">,
/// which can be extended to create a visitor which only needs to handle a subset
/// of the available methods.
/// \</summary>
/// \<typeparam name="Result">The return type of the visit operation.\</typeparam>
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <file.grammarName>BaseVisitor<Result> : AbstractParseTreeVisitor<Result>,
I<file.grammarName>Visitor<Result> {
    <file.visitorNames> { Iname |
    /// \<summary>
    <if(file.visitorLabelRuleNames.(Iname))>
    /// Visit a parse tree produced by the \<c><Iname>\</c>
    /// labeled alternative in \<see cref="<file.parserName>.<file.visitorLabelRuleNames.(Iname)">/.
    <else>
    /// Visit a parse tree produced by \<see cref="<file.parserName>.<Iname">"/>.
    <endif>
    /// \<para>
    /// The default implementation returns the result of calling \<see
    cref="AbstractParseTreeVisitor{Result}.VisitChildren(IRuleNode)">
    /// on \<paramref name="context">.
    /// \</para>
    /// \</summary>
    /// \<param name="context">The parse tree.\</param>
    /// \<return>The visitor result.\</return>
    public virtual Result Visit<Iname; format="cap">([NotNull] <csIdentifier.(file.parserName)>.<Iname;
    format="cap">Context context) { return VisitChildren(context); \}}; separator="\n"
    }
    <if(file.genPackage)>
    } // namespace <file.genPackage>
    <endif>
    >>

fileHeader(grammarFileName, ANTLRVersion) ::= <<
//-----
// \<auto-generated>
//   This code was generated by a tool.
//   ANTLR Version: <ANTLRVersion>
//
//   Changes to this file may cause incorrect behavior and will be lost if
//   the code is regenerated.
// \</auto-generated>
//-----

// Generated from <grammarFileName> by ANTLR <ANTLRVersion>

```

```

// Unreachable code detected
#pragma warning disable 0162
// The variable '...' is assigned but its value is never used
#pragma warning disable 0219
// Missing XML comment for publicly visible type or member '...'
#pragma warning disable 1591
// Ambiguous reference in cref attribute
#pragma warning disable 419

>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]
[System.CLSCompliant(false)]
public partial class <csIdentifier.(parser.name)> : <superClass>; null="Parser"> {
protected static DFA[] decisionToDFA;
protected static PredictionContextCache sharedContextCache = new PredictionContextCache();
<if(parser.tokens)>
public const int
<parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator=", ", wrap, anchor>;
<endif>
public const int
<parser.rules:{r | RULE_<r.name> = <r.index>}; separator=", ", wrap, anchor>;
public static readonly string[] ruleNames = {
<parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};

<vocabulary(parser.literalNames, parser.symbolicNames)>

public override string GrammarFileName { get { return "<parser.grammarFileName>"; } }

public override string[] RuleNames { get { return ruleNames; } }

public override string SerializedAtn { get { return new string(_serializedATN); } }

static <csIdentifier.(parser.name)>() {
decisionToDFA = new DFA[_ATN.NumberOfDecisions];
for (int i = 0; i < _ATN.NumberOfDecisions; i++) {
decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
}
}

<namedActions.members>

```

```

<parser:(ctor)()>
<funcs; separator="\n">

<if(sempredFuncs)>
public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <parser.sempredFuncs.values:{ f}
case <f.ruleIndex>: return <f.name>_sempred((<f.ctxType>)_localctx, predIndex);}; separator="\n">
    }
    return true;
    }
<sempredFuncs.values; separator="\n">
<endif>

<atn>
}
>>

vocabulary(literalNames, symbolicNames) ::= <<
private static readonly string[] _LiteralNames = {
    <literalNames:{ t | <t>}; null="null", separator=" ", wrap, anchor>
};
private static readonly string[] _SymbolicNames = {
    <symbolicNames:{ t | <t>}; null="null", separator=" ", wrap, anchor>
};
public static readonly IVocabulary DefaultVocabulary = new Vocabulary(_LiteralNames, _SymbolicNames);

[NotNull]
public override IVocabulary Vocabulary
{
    get
    {
        return DefaultVocabulary;
    }
}
>>

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
public override void Action(RuleContext _localctx, int ruleIndex, int actionIndex) {
    switch (ruleIndex) {
        <recog.actionFuncs.values:{ f}
case <f.ruleIndex> : <f.name>_action(<if(!recog.modes)>(<f.ctxType>)<endif>_localctx, actionIndex); break;};
separator="\n">
    }
    }
<actionFuncs.values; separator="\n">
<endif>

```

```

<if(sempredFuncs)>
public override bool Sempred(RuleContext _localctx, int ruleIndex, int predIndex) {
    switch (ruleIndex) {
        <recog.sempredFuncs.values: { f |
case <f.ruleIndex> : return <f.name>_sempred(<if(!recog.modes)><f.ctxType><endif>_localctx, predIndex);};
separator="\n">
    }
    return true;
}
<sempredFuncs.values; separator="\n">
<endif>
>>

```

```

parser_ctor(parser) ::= <<
    public <csIdentifier.(parser.name)>(ITokenStream input) : this(input, Console.Out, Console.Error) { }

    public <csIdentifier.(parser.name)>(ITokenStream input, TextWriter output, TextWriter errorOutput)
    : base(input, output, errorOutput)
    {
        Interpreter = new ParserATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
    }
>>

```

```

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleActionFunction(r, actions) ::= <<
private void <r.name>_action(<r.ctxType> _localctx, int actionIndex) {
    switch (actionIndex) {
        <actions: { index |
case <index>: <actions.(index)> break; }; separator="\n">
    }
}
>>

```

```

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/
RuleSempredFunction(r, actions) ::= <<
private bool <r.name>_sempred(<r.ctxType> _localctx, int predIndex) {
    switch (predIndex) {
        <actions: { index |
case <index>: return <actions.(index)>; }; separator="\n">
    }
    return true;
}
>>

```



```
RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
```

```
::= <<
```

```
<if(ruleCtx)>
```

```
<ruleCtx>
```

```
<endif>
```

```
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">
```

```
[RuleVersion(<namedActions.version; null="0">)]
```

```
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
```

```
<csIdentifier(currentRule.name)><(args; separator=", "> {
```

```
<currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
```

```
<csIdentifier(a.name)>>>);
```

```
EnterRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>);
```

```
<namedActions.init>
```

```
<locals; separator="\n">
```

```
try {
```

```
<if(currentRule.hasLookaheadBlock)>
```

```
int _alt;
```

```
<endif>
```

```
<code>
```

```
<postamble; separator="\n">
```

```
<namedActions.after>
```

```
}
```

```
<if(exceptions)>
```

```
<exceptions; separator="\n">
```

```
<else>
```

```
catch (RecognitionException re) {
```

```
_localctx.exception = re;
```

```
ErrorHandler.ReportError(this, re);
```

```
ErrorHandler.Recover(this, re);
```

```
}
```

```
<endif>
```

```
finally {
```

```
<finallyAction>
```

```
ExitRule();
```

```
}
```

```
return _localctx;
```

```
}
```

```
>>
```

```
LeftFactoredRuleFunction(currentRule,args,code,locals,namedActions,finallyAction,postamble) ::=
```

```
<<
```

```
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>private <endif><currentRule.ctxType>
```

```
<csIdentifier(currentRule.name)><(args; separator=", "> {
```

```
<currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, State<currentRule.args:{a | ,
```

```
<csIdentifier(a.name)>>>);
```



```

    ErrorHandler.Recover(this, re);
}
finally {
    <finallyAction>
    ExitRule();
}
return _localctx;
}
>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

```

```

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

```

```

[RuleVersion(<namedActions.version; null="0">)]
<if(currentRule.modifiers)><currentRule.modifiers:{f | <f> }><else>public <endif><currentRule.ctxType>
<csIdentifier.(currentRule.name)>(<args; separator=", ">) {
    return <csIdentifier.(currentRule.name)>(0<currentRule.args:{a | , <csIdentifier.(a.name)>}>>);
}

```

```

private <currentRule.ctxType> <csIdentifier.(currentRule.name)>(int _p<args:{a | , <a>}>) {
    ParserRuleContext _parentctx = Context;
    int _parentState = State;
    <currentRule.ctxType> _localctx = new <currentRule.ctxType>(Context, _parentState<currentRule.args:{a | ,
    <csIdentifier.(a.name)>}>>);
    <currentRule.ctxType> _prevctx = _localctx;
    int _startState = <currentRule.startState>;
    EnterRecursionRule(_localctx, <currentRule.startState>, RULE_<currentRule.name>, _p);
    <namedActions.init>
    <locals; separator="\n">
    try {
    <if(currentRule.hasLookaheadBlock)>
        int _alt;
    <endif>
    <code>
    <postamble; separator="\n">
    <namedActions.after>
    }
    catch (RecognitionException re) {
        _localctx.exception = re;
        ErrorHandler.ReportError(this, re);
        ErrorHandler.Recover(this, re);
    }
    finally {
        <finallyAction>

```

```

    UnrollRecursionContexts(_parentctx);
  }
  return _localctx;
}
>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>_localctx = new <currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(_localctx);<endif>
EnterOuterAlt(_localctx, <currentOuterMostAltCodeBlock.alt.altNum>);
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
{
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
}
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>
<preamble; separator="\n">
switch (TokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
<alt>
break;}; separator="\n">
default:
<error>
}
>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
switch (TokenStream.LA(1)) {
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
<alt>
break;}; separator="\n">
default:
break;
}
>>

```

```

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<

```

```

State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<preamble; separator="\n">
if (<expr>) {
  <alts; separator="\n">
}
<!else if ( !(<followExpr> ) <error>!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<preamble; separator="\n">
while (<loopExpr>) {
  <alts; separator="\n">
  State = <choice.loopBackStateNumber>;
  ErrorHandler.Sync(this);
  <iteration>
}
>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
State = <choice.blockStartStateNumber>;<! alt block decision !>
ErrorHandler.Sync(this);
<preamble; separator="\n">
do {
  <alts; separator="\n">
  State = <choice.stateNumber>;<! loopback/exit decision !>
  ErrorHandler.Sync(this);
  <iteration>
} while ( <loopExpr> );
>>

```

```

// LL(*) stuff

```

```

AltBlock(choice, preamble, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
<if(choice.label)><labelref(choice.label)> = TokenStream.LT(1);<endif>
<preamble; separator="\n">
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {
  <alts:{alt |
  case <i>:
    <alt>
    break;}; separator="\n">
}
>>

```

```

OptionalBlock(choice, alts, error) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
switch ( Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context) ) {
<alts:{ alt |
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
}
>>

```

```

StarBlock(choice, alts, sync, iteration) ::= <<
State = <choice.stateNumber>;
ErrorHandler.Sync(this);
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
while ( _alt!=<choice.exitAlt> && _alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER ) {
if ( _alt==1<if(!choice.ast.greedy)>+1<endif> ) {
<iteration>
<alts> <! should only be one !>
}
State = <choice.loopBackStateNumber>;
ErrorHandler.Sync(this);
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
}
>>

```

```

PlusBlock(choice, alts, error) ::= <<
State = <choice.blockStartStateNumber>;<! alt block decision !>
ErrorHandler.Sync(this);
_alt = 1<if(!choice.ast.greedy)>+1<endif>;
do {
switch ( _alt ) {
<alts:{ alt|
case <i><if(!choice.ast.greedy)>+1<endif>:
<alt>
break;}; separator="\n">
default:
<error>
}
State = <choice.loopBackStateNumber>;<! loopback/exit decision !>
ErrorHandler.Sync(this);
_alt = Interpreter.AdaptivePredict(TokenStream,<choice.decision>,Context);
} while ( _alt!=<choice.exitAlt> && _alt!=global::Antlr4.Runtime.Atn.ATN.INVALID_ALT_NUMBER );
>>

```

```

Sync(s) ::= "Sync(<s.expecting.name>);"

```

```

ThrowNoViableAlt(t) ::= "throw new NoViableAltException(this);"

```

```

TestSetInline(s) ::= <<
<s.bitsets:{bits | <if(rest(rest(bits.ttypes)))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s,
bits)><endif>}; separator=" || ">
>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test
testShiftInRange(shiftAmount) ::= <<
((<shiftAmount>) & ~0x3f) == 0
>>

// produces smaller bytecode only when bits.ttypes contains more than two items
bitsetBitfieldComparison(s, bits) ::= <%
(<testShiftInRange({<offsetShift(s.varName, bits.shift)>})> && ((1L \<< <offsetShift(s.varName, bits.shift)>) &
(<bits.ttypes:{ttype | (1L \<< <offsetShift(tokenType.(ttype), bits.shift)>}); separator=" | ">)) != 0)
%>

isZero ::= [
"0":true,
default:false
]

offsetShift(shiftAmount, offset) ::= <%
<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>
%>

// produces more efficient bytecode when bits.ttypes contains at most two items
bitsetInlineComparison(s, bits) ::= <%
<bits.ttypes:{ttype | <s.varName>==<tokenType.(ttype)>}; separator=" || ">
%>

cases(ttypes) ::= <<
<ttypes:{t | case <tokenType.(t)>:}; separator="\n">
>>

InvokeRule(r, argExprsChunks) ::= <<
State = <r.stateNumber>; <if(r.labels)><r.labels:{l | <labelref(l)> =
}><endif><csIdentifier.(r.name)><(if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)><endif><endif><arg
ExprsChunks>;
>>

MatchToken(m) ::= <<
State = <m.stateNumber>; <if(m.labels)><m.labels:{l | <labelref(l)> = }><endif>Match(<tokenType.(m.name)>);
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

```

```

CommonSetStuff(m, expr, capture, invert) ::= <<
State = <m.stateNumber>;
<if(m.labels)><m.labels:{1 | <labelref(l)> = }>TokenStream.LT(1);<endif>
<capture>
if ( <if(invert)><m.varName> \<= 0 || <else>!<endif>(<expr> ) ) {
  <if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>ErrorHandler.RecoverInline(this);
}
else {
  ErrorHandler.ReportMatch(this);
  Consume();
}
>>

Wildcard(w) ::= <<
State = <w.stateNumber>;
<if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>MatchWildcard();
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
State = <p.stateNumber>;
if (!(<chunks>)) throw new FailedPredicateException(this, <p.predicate><if(failChunks)>,
<failChunks><elseif(p.msg)>, <p.msg><endif>);
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
  <catchAction>
}
>>

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "Skip();"
LexerMoreCommand() ::= "More();"
LexerPopModeCommand() ::= "PopMode();"

LexerTypeCommand(arg, grammar) ::= "_type = <tokenType.(arg)>";
LexerChannelCommand(arg, grammar) ::= "_channel = <channelName.(arg)>";
LexerModeCommand(arg, grammar) ::= "_mode = <modeName.(arg)>";
LexerPushModeCommand(arg, grammar) ::= "PushMode(<modeName.(arg)>";

```



```

ActionText(t) ::= "<t.text>"
ActionTemplate(t) ::= "<t.st>"
ArgRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
LocalRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
RetValRef(a) ::= "_localctx.<csIdentifier.(a.name)>"
QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<csIdentifier.(a.name)>"
/** How to translate $tokenLabel */
TokenRef(t) ::= "<ctx(t)>.<csIdentifier.(tokenType.(t.name))>"
LabelRef(t) ::= "<ctx(t)>.<csIdentifier.(t.name)>"
ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(csIdentifier.(t.name))>"
SetAttr(s,rhsChunks) ::= "<ctx(s)>.<csIdentifier.(s.name)> = <rhsChunks>;"

TokenLabelType() ::= "<file.TokenLabelType; null={IToken}>"
InputSymbolType() ::= "<file.InputSymbolType; null={IToken}>"

TokenPropertyRef_text(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Text:null)"
TokenPropertyRef_type(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Type:0)"
TokenPropertyRef_line(t) ::= "<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Line:0)"
TokenPropertyRef_pos(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.CharPositionInLine:0)"
TokenPropertyRef_channel(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.Channel:0)"
TokenPropertyRef_index(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?<ctx(t)>.<tokenType.(t.label)>.TokenIndex:0)"
TokenPropertyRef_int(t) ::=
"<ctx(t)>.<tokenType.(t.label)>!=null?int.Parse(<ctx(t)>.<tokenType.(t.label)>.Text):0)"

RulePropertyRef_start(r) ::= "<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Start):null)"
RulePropertyRef_stop(r) ::= "<ctx(r)>.<r.label>!=null?(<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_text(r) ::=
"<ctx(r)>.<r.label>!=null?TokenStream.GetText(<ctx(r)>.<r.label>.Start,<ctx(r)>.<r.label>.Stop):null)"
RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"
RulePropertyRef_parser(r) ::= "this"

ThisRulePropertyRef_start(r) ::= "_localctx.Start"
ThisRulePropertyRef_stop(r) ::= "_localctx.Stop"
ThisRulePropertyRef_text(r) ::= "TokenStream.GetText(_localctx.Start, TokenStream.LT(-1))"
ThisRulePropertyRef_ctx(r) ::= "_localctx"
ThisRulePropertyRef_parser(r) ::= "this"

NonLocalAttrRef(s) ::= <%((<s.ruleName;
format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)>%>
SetNonLocalAttr(s, rhsChunks) ::=
<%((<s.ruleName; format="cap">Context)GetInvokingContext(<s.ruleIndex>)).<csIdentifier.(s.name)> =
<rhsChunks>;%>

AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.Add(<labelref(a.label)>);"

```

```

TokenDecl(t) ::= "<TokenLabelType() <csIdentifier.(tokenType.(t.name))>"
TokenTypeDecl(t) ::= "int <csIdentifier.(tokenType.(t.name))>";"
TokenListDecl(t) ::= "IList<IToken> <csIdentifier.(tokenType.(t.name))> = new List<IToken>()"
RuleContextDecl(r) ::= "<r.ctxName> <csIdentifier.(r.name)>"
RuleContextListDecl(rdecl) ::= "IList<<rdecl.ctxName>> <csIdentifier.(rdecl.name)> = new
List<<rdecl.ctxName>>()"

```

```

contextGetterCollection(elementType) ::= <%
<elementType>[]
%>

```

```

ContextTokenGetterDecl(t) ::=
    "public ITerminalNode <csIdentifier.(tokenType.(t.name))>() { return
    GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, 0); }"
ContextTokenListGetterDecl(t) ::= <<
    public <contextGetterCollection("ITerminalNode")> <csIdentifier.(tokenType.(t.name))>() { return
    GetTokens(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>); }
    >>
ContextTokenListIndexedGetterDecl(t) ::= <<
    public ITerminalNode <csIdentifier.(tokenType.(t.name))>(int i) {
    return GetToken(<csIdentifier.(parser.name)>.<csIdentifier.(tokenType.(t.name))>, i);
    }
    >>
ContextRuleGetterDecl(r) ::= <<
    public <r.ctxName> <csIdentifier.(r.name)>() {
    return GetRuleContext<<r.ctxName>>()();
    }
    >>
ContextRuleListGetterDecl(r) ::= <<
    public <contextGetterCollection({<r.ctxName>})> <csIdentifier.(r.name)>() {
    return GetRuleContexts<<r.ctxName>>()();
    }
    >>
ContextRuleListIndexedGetterDecl(r) ::= <<
    public <r.ctxName> <csIdentifier.(r.name)>(int i) {
    return GetRuleContext<<r.ctxName>>(i);
    }
    >>

```

```

LexerRuleContext() ::= "RuleContext"

```

```

/** The rule context name is the rule followed by a suffix; e.g.,
* r becomes rContext.
*/

```

```

RuleContextNameSuffix() ::= "Context"

```

```

ImplicitTokenLabel(tokenName) ::= "_<tokenType.(tokenName)>"

```

```

ImplicitRuleLabel(ruleName) ::= "_<ruleName>"

```

```
ImplicitsetLabel(id) ::= "_tset<id>"
ListLabelName(label) ::= "_<label>"
```

```
CaptureNextToken(d) ::= "<d.varName> = TokenStream.LT(1);"
CaptureNextTokenType(d) ::= "<d.varName> = TokenStream.LA(1);"
```

```
StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers,
    superClass={ParserRuleContext}) ::= <<
public partial class <struct.name> :
<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif><if(interfaces)>, <interfaces>
separator=", "><endif> {
<attrs:{a | public <a>;}; separator="\n">
<getters:{g | <g>;}; separator="\n">
<if(ctorAttrs)>public <struct.name>(ParserRuleContext parent, int invokingState) : base(parent, invokingState) {
}<endif>
public <struct.name>(ParserRuleContext parent, int invokingState<ctorAttrs:{a | , <a>>)>
: base(parent, invokingState)
{
<struct.ctorAttrs:{a | this.<csIdentifier.(a.name)> = <csIdentifier.(a.name)>;}; separator="\n">
}
public override int RuleIndex { get { return RULE_<struct.derivedFromName>; } }
<if(struct.provideCopyFrom)><! don't need copy unless we have subclasses !>
public <struct.name>() { }
public virtual void CopyFrom(<struct.name> context) {
base.CopyFrom(context);
<struct.attrs:{a | this.<csIdentifier.(a.name)> = context.<csIdentifier.(a.name)>;}; separator="\n">
}
<endif>
<dispatchMethods; separator="\n">
<extensionMembers; separator="\n">
}
>>
```

```
AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
public partial class <struct.name> : <currentRule.name; format="cap">Context {
<attrs:{a | public <a>;}; separator="\n">
<getters:{g | <g>;}; separator="\n">
public <struct.name>(<currentRule.name; format="cap">Context context) { CopyFrom(context); }
<dispatchMethods; separator="\n">
}
>>
```

```
ListenerDispatchMethod(method) ::= <<
public override void <if(method.isEnter)>Enter<else>Exit<endif>Rule(IParseTreeListener listener) {
I<parser.grammarName>Listener typedListener = listener as I<parser.grammarName>Listener;
if (typedListener != null) typedListener.<if(method.isEnter)>Enter<else>Exit<endif><struct.derivedFromName>;
format="cap">(this);
}
}
```

>>

```
VisitorDispatchMethod(method) ::= <<
public override TResult Accept<TResult>(IParseTreeVisitor<TResult> visitor) {
    I<parser.grammarName>Visitor<TResult> typedVisitor = visitor as I<parser.grammarName>Visitor<TResult>;
    if (typedVisitor != null) return typedVisitor.Visit<struct.derivedFromName; format="cap">(this);
    else return visitor.VisitChildren(this);
}
>>
```

```
AttributeDecl(d) ::= "<d.type> <csIdentifier.(d.name)><if(d.initValue)> = <d.initValue><endif>"
```

```
/** If we don't know location of label def x, use this template */
```

```
labelref(x) ::= "<if(!x.isLocal)><typedContext(x.ctx)><endif><csIdentifier.(x.name)>"
```

```
/** For any action chunk, what is correctly-typed context struct ptr? */
```

```
ctx(actionChunk) ::= "<typedContext(actionChunk.ctx)>"
```

```
// only casts _localctx to the type when the cast isn't redundant (i.e. to a sub-context for a labeled alt)
```

```
typedContext(ctx) ::= "<if(ctx.provideCopyFrom)>((<ctx.name>)_localctx)<else>_localctx<endif>"
```

```
// used for left-recursive rules
```

```
recRuleAltPredicate(ruleName,opPrec) ::= "Precpred(Context, <opPrec>)"
```

```
recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>;"
```

```
recRuleSetStopToken() ::= "Context.Stop = TokenStream.LT(-1);"
```

```
recRuleAltStartAction(ruleName, ctxName, label, isListLabel) ::= <<
```

```
_localctx = new <ctxName>Context(_parentctx, _parentState);
```

```
<if(label)>
```

```
<if(isListLabel)>
```

```
_localctx.<label>.Add(_prevctx);
```

```
<else>
```

```
_localctx.<label> = _prevctx;
```

```
<endif>
```

```
<endif>
```

```
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
```

```
>>
```

```
recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
```

```
_localctx = new <currentAltLabel; format="cap">Context(new <ruleName; format="cap">Context(_parentctx,
_parentState));
```

```
<if(label)>
```

```
<if(isListLabel)>
```

```
((<currentAltLabel; format="cap">Context)_localctx).<label>.Add(_prevctx);
```

```
<else>
```

```
((<currentAltLabel; format="cap">Context)_localctx).<label> = _prevctx;
```

```
<endif>
```

```
<endif>
```

```
PushNewRecursionContext(_localctx, _startState, RULE_<ruleName>);
```

```
>>
```

```
recRuleReplaceContext(ctxName) ::= <<  
_localctx = new <ctxName>Context(_localctx);  
Context = _localctx;  
_prevctx = _localctx;  
>>
```

```
recRuleSetPrevCtx() ::= <<  
if ( ParseListeners!=null )  
    TriggerExitRuleEvent();  
_prevctx = _localctx;  
>>
```

```
LexerFile(file, lexer, namedActions) ::= <<  
<fileHeader(file.grammarFileName, file.ANTLRVersion)>  
<if(file.genPackage)>  
namespace <file.genPackage> {  
<endif>  
<namedActions.header>  
using System;  
using System.IO;  
using System.Text;  
using Antlr4.Runtime;  
using Antlr4.Runtime.Atn;  
using Antlr4.Runtime.Misc;  
using DFA = Antlr4.Runtime.Dfa.DFA;  
  
<lexer>  
<if(file.genPackage)>  
} // namespace <file.genPackage>  
<endif>  
>>
```

```
Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<  
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<file.ANTLRVersion>")]  
[System.CLSCompliant(false)]  
public partial class <csIdentifier.(lexer.name)> : <superClass; null="Lexer"> {  
    protected static DFA[] decisionToDFA;  
    protected static PredictionContextCache sharedContextCache = new PredictionContextCache();  
    public const int  
        <lexer.tokens:{k | <tokenType.(k)>=<lexer.tokens.(k)>}; separator=", ", wrap, anchor>;  
    <if(lexer.channels)>  
    public const int  
        <lexer.channels:{k | <csIdentifier.(k)>=<lexer.channels.(k)>}; separator=", ", wrap, anchor>;  
    <endif>
```

```

<if(rest(lexer.modes))>
public const int
<rest(lexer.modes):{m | <m>=<i>}; separator=", ", wrap, anchor>;
<endif>
public static string[] channelNames = {
"DEFAULT_TOKEN_CHANNEL", "HIDDEN" <if (lexer.channels)>, <lexer.channels:{c | "<c>"}; separator=", ",
wrap, anchor><endif>
};

public static string[] modeNames = {
<lexer.modes:{m | "<m>"}; separator=", ", wrap, anchor>
};

public static readonly string[] ruleNames = {
<lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor>
};

<namedActions.members>

public <csIdentifier.(lexer.name)>(ICharStream input)
: this(input, Console.Out, Console.Error) { }

public <csIdentifier.(lexer.name)>(ICharStream input, TextWriter output, TextWriter errorOutput)
: base(input, output, errorOutput)
{
Interpreter = new LexerATNSimulator(this, _ATN, decisionToDFA, sharedContextCache);
}

<vocabulary(lexer.literalNames, lexer.symbolicNames)>

public override string GrammarFileName { get { return "<lexer.grammarFileName>"; } }

public override string[] RuleNames { get { return ruleNames; } }

public override string[] ChannelNames { get { return channelNames; } }

public override string[] ModeNames { get { return modeNames; } }

public override string SerializedAtn { get { return new string(_serializedATN); } }

static <csIdentifier.(lexer.name)>() {
decisionToDFA = new DFA[_ATN.NumberOfDecisions];
for (int i = 0; i < _ATN.NumberOfDecisions; i++) {
decisionToDFA[i] = new DFA(_ATN.GetDecisionState(i), i);
}
}
<dumpActions(lexer, "", actionFuncs, sempredFuncs)>
<atn>

```

```
}  
>>
```

```
SerializedATN(model) ::= <<  
private static char[] _serializedATN = {  
  <model.serialized; separator=",", wrap>,  
};  
  
public static readonly ATN _ATN =  
  new ATNDeserializer().Deserialize(_serializedATN);
```

```
>>
```

```
initValue(typeName) ::= <<  
default(<typeName>)  
>>
```

```
codeFileExtension() ::= ".cs"
```

```
modeName ::= [  
  "DEFAULT_MODE" : "DefaultMode",  
  default : key  
]
```

```
channelName ::= [  
  "HIDDEN" : "Hidden",  
  "DEFAULT_TOKEN_CHANNEL" : "DefaultTokenChannel",  
  default : key  
]
```

```
tokenType ::= [  
  "EOF" : "Eof",  
  default : key  
]
```

```
csIdentifier ::= [  
  "abstract" : "@abstract",  
  "as" : "@as",  
  "base" : "@base",  
  "bool" : "@bool",  
  "break" : "@break",  
  "byte" : "@byte",  
  "case" : "@case",  
  "catch" : "@catch",  
  "char" : "@char",  
  "checked" : "@checked",
```

"class" : "@class",
"const" : "@const",
"continue" : "@continue",
"decimal" : "@decimal",
"default" : "@default",
"delegate" : "@delegate",
"do" : "@do",
"double" : "@double",
"else" : "@else",
"enum" : "@enum",
"event" : "@event",
"explicit" : "@explicit",
"extern" : "@extern",
"false" : "@false",
"finally" : "@finally",
"fixed" : "@fixed",
"float" : "@float",
"for" : "@for",
"foreach" : "@foreach",
"goto" : "@goto",
"if" : "@if",
"implicit" : "@implicit",
"in" : "@in",
"int" : "@int",
"interface" : "@interface",
"internal" : "@internal",
"is" : "@is",
"lock" : "@lock",
"long" : "@long",
"namespace" : "@namespace",
"new" : "@new",
"null" : "@null",
"object" : "@object",
"operator" : "@operator",
"out" : "@out",
"override" : "@override",
"params" : "@params",
"private" : "@private",
"protected" : "@protected",
"public" : "@public",
"readonly" : "@readonly",
"ref" : "@ref",
"return" : "@return",
"sbyte" : "@sbyte",
"sealed" : "@sealed",
"short" : "@short",
"sizeof" : "@sizeof",
"stackalloc" : "@stackalloc",


```
"static" : "@static",
"string" : "@string",
"struct" : "@struct",
"switch" : "@switch",
"this" : "@this",
"throw" : "@throw",
>true" : "@true",
"try" : "@try",
"typeof" : "@typeof",
"uint" : "@uint",
"ulong" : "@ulong",
"unchecked" : "@unchecked",
"unsafe" : "@unsafe",
"ushort" : "@ushort",
"using" : "@using",
"virtual" : "@virtual",
"values" : "@values",
"void" : "@void",
"volatile" : "@volatile",
"while" : "@while",
default : key
]
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/codegen/CSharp/CSharp.stg
```

No license file was found, but licenses were detected in source scan.

```
* [The "BSD license"]
```

```
* All rights reserved.
```

```
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
```

- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * 3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

```
[The "BSD licence"]
```

```
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:
```

- 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- 2. Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ANTLRParser.g
No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

lexer grammar ActionSplitter;

options { filter=true; }

@header {

package org.antlr.v4.parse;

import org.antlr.v4.tool.*;

import org.antlr.v4.tool.ast.*;

}

```

@members {
ActionSplitterListener delegate;

public ActionSplitter(CharStream input, ActionSplitterListener delegate) {
    this(input, new RecognizerSharedState());
    this.delegate = delegate;
}

/** force filtering (and return tokens). triggers all above actions. */
public List<Token> getActionTokens() {
    List<Token> chunks = new ArrayList<Token>();
    Token t = nextToken();
    while ( t.getType()!=Token.EOF ) {
        chunks.add(t);
        t = nextToken();
    }
    return chunks;
}

private boolean isIDStartChar(int c) {
    return c == '_' || Character.isLetter(c);
}

}

// ignore comments right away

COMMENT
: /*' ( options {greedy=false;} : . )* */' {delegate.text($text);}
;

LINE_COMMENT
: //' ~(\n|\r)* \r? \n' {delegate.text($text);}
;

SET_NONLOCAL_ATTR
: '$ x=ID ':' y=ID WS? '=' expr=ATTR_VALUE_EXPR ';'
{
    delegate.setNonLocalAttr($text, $x, $y, $expr);
}
;

NONLOCAL_ATTR
: '$ x=ID ':' y=ID {delegate.nonLocalAttr($text, $x, $y);}
;

QUALIFIED_ATTR
: '$ x=ID '.' y=ID {input.LA(1)!=''}? {delegate.qualifiedAttr($text, $x, $y);}

```

```

;

SET_ATTR
: '$ x=ID WS? '=' expr=ATTR_VALUE_EXPR ';'
{
    delegate.setAttr($text, $x, $expr);
}
;

ATTR
: '$ x=ID { delegate.attr($text, $x); }
;

// Anything else is just random text
TEXT
@init {StringBuilder buf = new StringBuilder();}
@after {delegate.text(buf.toString());}
: ( c=~(\\| '$) {buf.append((char)$c);}
| '\\$' {buf.append('$);}
| '\\ c=~('$) {buf.append('\\').append((char)$c);}
| {!isIDStartChar(input.LA(2))}? => '$' {buf.append('$);}
)+
;

fragment
ID : ('a'..'z'|'A'..'Z'|'_') ('a'..'z'|'A'..'Z'|'0'..'9'|'_')*
;

/** Don't allow an = as first char to prevent $x == 3; kind of stuff. */
fragment
ATTR_VALUE_EXPR
: ~'=' (~';')*
;

fragment
WS : ('\\t'|'\\n'|'\\r')+
;

Found in path(s):
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ActionSplitter.g
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* All rights reserved.
*

```

- * Redistribution and use in source and binary forms, with or without
- * modification, are permitted provided that the following conditions
- * are met:
- *
- * 1. Redistributions of source code must retain the above copyright
- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
- * notice, this list of conditions and the following disclaimer in the
- * documentation and/or other materials provided with the distribution.
- * 3. The name of the author may not be used to endorse or promote products
- * derived from this software without specific prior written permission.
- *
- * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
- * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
- * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
- * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
- * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
- * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
- * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
- * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
- * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
- * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- */

```
atn(startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph ATN {
rankdir=LR;
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
dfa(name, startState, states, edges, rankdir, decisionRanks, useBox) ::= <<
digraph <name> {
<if(rankdir)>rankdir=<rankdir>;<endif>
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
decision-rank(states) ::= <<
{rank=same; rankdir=TB; <states:{s | s<s>}; separator=" "; ">}
>>
```

```
edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>:p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
```

```

label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>;
>>

action-edge(src,target,label,arrowhead,transitionIndex) ::= <<
<src><if(transitionIndex)>p<transitionIndex><endif> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7,
label = "<label>"<if(arrowhead)>, arrowhead = <arrowhead><endif>;
>>

epsilon-edge(src,label,target,arrowhead,transitionIndex,loopback=false) ::= <<
<src><if(transitionIndex)>p<transitionIndex><endif> -> <target> [fontname="Times-Italic",
label="&epsilon;"<if(loopback)>, style="dashed"<endif>;
>>

state(state, label, name, transitions) ::= <%
<name>[fontsize=11,
label="
  <! rest(transition) tests for decision states: these nodes have a non-empty set of transitions after the first one. !>
  <if(rest(transitions))>
    {
      <! Label on the left side of the record node. !>
      <label>
      |
      <! Named ports in order on right side of record node, no display text. !>
      {<transitions:{t|\<p<i0>>}; separator="|">}
    }
  <else>
    <label>
  <endif>
  "
  <if(rest(transitions))>
    , shape=record, fixedsize=false
  <else>
    , shape=circle, fixedsize=true, width=.55
  <endif>
  , peripheries=1];
%>

stopstate(name,label,actionIndex,useBox) ::= <<
<name>[fontsize=11, label="<label>"<if(actionIndex)>,\naction:<actionIndex><endif>",
<if(useBox)>shape=polygon,sides=4,peripheries=2,fixedsize=false<else>shape=doublecircle, fixedsize=true,
width=.6<endif>];
>>

Found in path(s):
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/tool/templates/dot/graphs.stg
No license file was found, but licenses were detected in source scan.

/*

```

[The "BSD license"]
Copyright (c) 2011 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/GrammarTreeVisitor.java

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]
* Copyright (c) 2012-2016 Terence Parr
* Copyright (c) 2012-2016 Sam Harwell
* Copyright (c) 2014 Eric Vergnaud
* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the

```

* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** ANTLR tool checks output templates are compatible with tool code generation.
* For now, a simple string match used on x.y of x.y.z scheme.
* Must match Tool.VERSION during load to templates.
*
* REQUIRED.
*/

```

```

pythonTypeInitMap ::= [
  "bool":"False",
  "int":"0",
  "float":"0.0",
  "str": "",
  default:"None" // anything other than a primitive type is an object
]

```

```

// args must be <object-model-object>, <fields-resulting-in-STs>

```

```

ParserFile(file, parser, namedActions, contextSuperClass) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
# encoding: utf-8
from antlr4 import *
from io import StringIO
from typing.io import TextIO
import sys

<namedActions.header>
<parser>

>>

```

```

ListenerFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>

```



```

from antlr4 import *
if __name__ is not None and "." in __name__:
    from <file.parserName> import <file.parserName>
else:
    from <file.parserName> import <file.parserName>
<header>

# This class defines a complete listener for a parse tree produced by <file.parserName>.
class <file.grammarName>Listener(ParseTreeListener):

    <file.listenerNames:{lname |
# Enter a parse tree produced by <file.parserName>#<lname>.
def enter<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
    pass

# Exit a parse tree produced by <file.parserName>#<lname>.
def exit<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
    pass

}; separator="\n">

>>

```

```

VisitorFile(file, header, namedActions) ::= <<
<fileHeader(file.grammarFileName, file.ANTLRVersion)>
from antlr4 import *
if __name__ is not None and "." in __name__:
    from <file.parserName> import <file.parserName>
else:
    from <file.parserName> import <file.parserName>
<header>

# This class defines a complete generic visitor for a parse tree produced by <file.parserName>.

class <file.grammarName>Visitor(ParseTreeVisitor):

    <file.visitorNames:{lname |
# Visit a parse tree produced by <file.parserName>#<lname>.
def visit<lname; format="cap">(self, ctx:<file.parserName>.<lname; format="cap">Context):
    return self.visitChildren(ctx)

}; separator="\n">

del <file.parserName>
>>

```

```

fileHeader(grammarFileName, ANTLRVersion) ::= <<
# Generated from <grammarFileName> by ANTLR <ANTLRVersion>
>>

Parser(parser, funcs, atn, sempredFuncs, superClass) ::= <<
<Parser_(ctor="parser_ctor", ...)>
>>

Parser_(parser, funcs, atn, sempredFuncs, ctor, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
    from .<superClass> import <superClass>
else:
    from <superClass> import <superClass>

<endif>
<atn>

class <parser.name> ( <if(superClass)><superClass><else>Parser<endif> ):

    grammarFileName = "<parser.grammarFileName>"

    atn = ATNDeserializer().deserialize(serializedATN())

    decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

    sharedContextCache = PredictionContextCache()

    literalNames = [ <parser.literalNames:{t | <t>}; null="\\"<INVALID>\\"", separator=", ", wrap, anchor > ]

    symbolicNames = [ <parser.symbolicNames:{t | <t>}; null="\\"<INVALID>\\"", separator=", ", wrap, anchor > ]

    <parser.rules:{r | RULE_<r.name> = <r.index>}; separator="\n", wrap, anchor >

    ruleNames = [ <parser.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor > ]

    EOF = <TokenLabelType().EOF

    <if(parser.tokens)>
    <parser.tokens:{k | <k>=<parser.tokens.(k)>}; separator="\n", wrap, anchor >
    <endif>

    <parser:(ctor)()>

    <namedActions.members>

    <funcs; separator="\n">

```

```

<if(sempredFuncs)>
  def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
    if self._predicates == None:
      self._predicates = dict()
<parser.sempredFuncs.values:{ f |
  self._predicates[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n
  " >
  pred = self._predicates.get(ruleIndex, None)
  if pred is None:
    raise Exception("No predicate with index:" + str(ruleIndex))
  else:
    return pred(localctx, predIndex)

  <sempredFuncs.values; separator="\n">
<endif>

```

```
>>
```

```

dumpActions(recog, argFuncs, actionFuncs, sempredFuncs) ::= <<
<if(actionFuncs)>
def action(self, localctx:RuleContext, ruleIndex:int, actionIndex:int):
  if self._actions is None:
    actions = dict()
<recog.actionFuncs.values:{ f |
  actions[<f.ruleIndex>] = self.<f.name>_action }; separator="\n">
  self._actions = actions
  action = self._actions.get(ruleIndex, None)
  if action is not None:
    action(localctx, actionIndex)
  else:
    raise Exception("No registered action for:" + str(ruleIndex))

  <actionFuncs.values; separator="\n">

<endif>

```

```

<if(sempredFuncs)>
def sempred(self, localctx:RuleContext, ruleIndex:int, predIndex:int):
  if self._predicates is None:
    preds = dict()
<recog.sempredFuncs.values:{ f |
  preds[<f.ruleIndex>] = self.<f.name>_sempred}; separator="\n">
  self._predicates = preds
  pred = self._predicates.get(ruleIndex, None)
  if pred is not None:
    return pred(localctx, predIndex)
  else:
    raise Exception("No registered predicate for:" + str(ruleIndex))

```

```

<sempredFuncs.values; separator="\n">
<endif>
>>

```

```

parser_ctor(p) ::= <<
def __init__(self, input:TokenStream, output:TextIO = sys.stdout):
    super().__init__(input, output)
    self.checkVersion("<file.ANTLRVersion>")
    self._interp = ParserATNSimulator(self, self.atn, self.decisionsToDFA, self.sharedContextCache)
    self._predicates = None

```

```

>>

```

```

/* This generates a private method since the actionIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleActionFunction(r, actions) ::= <<

```

```

def <r.name>_action(self, localctx:<r.ctxType> , actionIndex:int):
<actions:{index|
<if(first(actions))>
    if actionIndex == <index>:
        <actions.(index)>
<elseif(rest(actions))>
    elif actionIndex == <index>:
        <actions.(index)>
<endif> }; separator="\n">

```

```

>>

```

```

/* This generates a private method since the predIndex is generated, making an
* overriding implementation impossible to maintain.
*/

```

```

RuleSempredFunction(r, actions) ::= <<

```

```

def <r.name>_sempred(self, localctx:<r.ctxType>, predIndex:int):
    <actions:{index|
<if(first(actions))>
    if predIndex == <index>:
        return <actions.(index)>
<elseif(rest(actions))>
    elif predIndex == <index>:
        return <actions.(index)>
<endif> }; separator="\n">

```

```

>>

```

```

RuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,namedActions,finallyAction,postamble,exceptions)
::= <<

```

```

<ruleCtx>

<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self<currentRule.args:{a | , <a.name><if(a.type)><a.type><endif>}>):

    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, self.state<currentRule.args:{a | , <a.name>}>)
    self.enterRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>)
    <namedActions.init>
    <locals; separator="\n">
    try:
        <code>
        <postamble; separator="\n">
        <namedActions.after>
    <if(exceptions)>
    <exceptions; separator="\n">
    <else>
    except RecognitionException as re:
        localctx.exception = re
        self._errHandler.reportError(self, re)
        self._errHandler.recover(self, re)
    <endif>
    finally:
        <finallyAction>
        self.exitRule()
    return localctx

>>

```

```

LeftRecursiveRuleFunction(currentRule,args,code,locals,ruleCtx,altLabelCtxs,
    namedActions,finallyAction,postamble) ::=
<<

```

```

<ruleCtx>
<altLabelCtxs:{l | <altLabelCtxs.(l)>}; separator="\n">

def <currentRule.name>(self, _p:int=0<if(currentRule.args)>, <args:{a | , <a>}><endif>):
    _parentctx = self._ctx
    _parentState = self.state
    localctx = <parser.name>.<currentRule.ctxType>(self, self._ctx, _parentState<args:{a | , <a.name>}>)
    _prevctx = localctx
    _startState = <currentRule.startState>
    self.enterRecursionRule(localctx, <currentRule.startState>, self.RULE_<currentRule.name>, _p)
    <namedActions.init>
    <locals; separator="\n">
    try:
        <code>

```

```

    <postamble; separator="\n">
    <namedActions.after>
except RecognitionException as re:
    localctx.exception = re
    self._errHandler.reportError(self, re)
    self._errHandler.recover(self, re)
finally:
    <finallyAction>
    self.unrollRecursionContexts(_parentctx)
return localctx

>>

```

```

CodeBlockForOuterMostAlt(currentOuterMostAltCodeBlock, locals, preamble, ops) ::= <<
<if(currentOuterMostAltCodeBlock.altLabel)>localctx = <parser.name>.<currentOuterMostAltCodeBlock.altLabel;
format="cap">Context(self, localctx)<endif>
self.enterOuterAlt(localctx, <currentOuterMostAltCodeBlock.alt.altNum>)
<CodeBlockForAlt(currentAltCodeBlock=currentOuterMostAltCodeBlock, ...)>
>>

```

```

CodeBlockForAlt(currentAltCodeBlock, locals, preamble, ops) ::= <<
<locals; separator="\n">
<preamble; separator="\n">
<ops; separator="\n">
>>

```

```

LL1AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
    <alt>
    pass}; separator="\n1">
else:
    <error>

>>

```

```

LL1OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
token = self._input.LA(1)
<choice.altLook,alts:{look,alt| <cases(ttypes=look)>
    <alt>
    pass}; separator="\n1">
else:

```

```

    pass
  >>

LL1OptionalBlockSingleAlt(choice, expr, alts, preamble, error, followExpr) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
if <expr>:
    <alts; separator="\n">

<!else if ( !(<followExpr> ) <error>!>
>>

```

```

LL1StarBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<preamble; separator="\n">
while <loopExpr>:
    <alts; separator="\n">
    self.state = <choice.loopBackStateNumber>
    self._errHandler.sync(self)
    <iteration>

>>

```

```

LL1PlusBlockSingleAlt(choice, loopExpr, alts, preamble, iteration) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
<preamble; separator="\n">
while True:
    <alts; separator="\n">
    self.state = <choice.stateNumber> <! loopback/exit decision !>
    self._errHandler.sync(self)
    <iteration>
    if not (<loopExpr>):
        break

>>

```

```

// LL(*) stuff

```

```

AltBlock(choice, preamble, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
<if(choice.label)><labelref(choice.label)> = _input.LT(1)<endif>
<preamble; separator="\n">
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

```

```

<alts:{ alt |
if la_ == <i>:
  <alt>
  pass
}; separator="\nel">

```

```
>>
```

```

OptionalBlock(choice, alts, error) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
la_ = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
<alts:{ alt |
if la_ == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">

```

```
>>
```

```

StarBlock(choice, alts, sync, iteration) ::= <<
self.state = <choice.stateNumber>
self._errHandler.sync(self)
_alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  if _alt==1<if(!choice.ast.greedy)>+1<endif>:
    <iteration>
    <alts> <! should only be one !>
  self.state = <choice.loopBackStateNumber>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

```

```
>>
```

```

PlusBlock(choice, alts, error) ::= <<
self.state = <choice.blockStartStateNumber> <! alt block decision !>
self._errHandler.sync(self)
_alt = 1<if(!choice.ast.greedy)>+1<endif>
while _alt!=<choice.exitAlt> and _alt!=ATN.INVALID_ALT_NUMBER:
  <alts:{ alt |
if _alt == <i><if(!choice.ast.greedy)>+1<endif>:
  <alt>
}; separator="\nel">
  else:
    <error>
  self.state = <choice.loopBackStateNumber> <! loopback/exit decision !>
  self._errHandler.sync(self)
  _alt = self._interp.adaptivePredict(self._input,<choice.decision>,self._ctx)

```


>>

Sync(s) ::= "sync(<s.expecting.name>)"

ThrowNoViableAlt(t) ::= "raise NoViableAltException(self)"

TestSetInline(s) ::= <<

<s.bitsets:{ bits | <if(rest(rest(bits.ttypes))><bitsetBitfieldComparison(s, bits)><else><bitsetInlineComparison(s, bits)><endif>}; separator=" or ">

>>

// Java language spec 15.19 - shift operators mask operands rather than overflow to 0... need range test

testShiftInRange(shiftAmount) ::= <<

((<shiftAmount>) & ~0x3f) == 0

>>

// produces smaller bytecode only when bits.ttypes contains more than two items

bitsetBitfieldComparison(s, bits) ::= <%

(<testShiftInRange({ <offsetShiftVar(s.varName, bits.shift)>})> and ((1 <|< <offsetShiftVar(s.varName, bits.shift)>) & (<bits.ttypes:{ ttype | (1 <|< <offsetShiftType(ttype, bits.shift)>}); separator=" | ">) != 0)
%>

isZero ::= [

"0":true,

default:false

]

offsetShiftVar(shiftAmount, offset) ::= <%

<if(!isZero.(offset))><shiftAmount> - <offset><else><shiftAmount><endif>

%>

offsetShiftType(shiftAmount, offset) ::= <%

<if(!isZero.(offset))><parser.name>.<shiftAmount> - <offset><else><parser.name>.<shiftAmount><endif>

%>

// produces more efficient bytecode when bits.ttypes contains at most two items

bitsetInlineComparison(s, bits) ::= <%

<bits.ttypes:{ ttype | <s.varName>==<parser.name>.<ttype>}; separator=" or ">

%>

cases(ttypes) ::= <<

if token in [<ttypes:{ t | <parser.name>.<t>}; separator=", ">]:

>>

InvokeRule(r, argExprsChunks) ::= <<

self.state = <r.stateNumber>

<if(r.labels)><r.labels:{ l | <labelref(l)> =

}><endif>self.<r.name>(<if(r.ast.options.p)><r.ast.options.p><if(argExprsChunks)><endif><endif><argExprsChu

```

nks>)
>>

MatchToken(m) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{1 | <labelref(l)> = }><endif>self.match(<parser.name>.<m.name>)
>>

MatchSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, false)>"

MatchNotSet(m, expr, capture) ::= "<CommonSetStuff(m, expr, capture, true)>"

CommonSetStuff(m, expr, capture, invert) ::= <<
self.state = <m.stateNumber>
<if(m.labels)><m.labels:{1 | <labelref(l)> = }>self._input.LT(1)<endif>
<capture>
<if(invert)>if <m.varName> \<= 0 or <expr><else>if not(<expr>)<endif>:
<if(m.labels)><m.labels:{1 | <labelref(l)> = }><else> <endif>self._errHandler.recoverInline(self)
else:
    self._errHandler.reportMatch(self)
    self.consume()
>>

Wildcard(w) ::= <<
self.state = <w.stateNumber>
<if(w.labels)><w.labels:{1 | <labelref(l)> = }><endif>self.matchWildcard()
>>

// ACTION STUFF

Action(a, foo, chunks) ::= "<chunks>"

ArgAction(a, chunks) ::= "<chunks>"

SemPred(p, chunks, failChunks) ::= <<
self.state = <p.stateNumber>
if not <chunks>:
    from antlr4.error.Errors import FailedPredicateException
    raise FailedPredicateException(self, <p.predicate><if(failChunks)>, <failChunks><elseif(p.msg)>,
<p.msg><endif>)
>>

ExceptionClause(e, catchArg, catchAction) ::= <<
catch (<catchArg>) {
    <catchAction>
}
>>

```

// lexer actions are not associated with model objects

LexerSkipCommand() ::= "skip()"

LexerMoreCommand() ::= "more()"

LexerPopModeCommand() ::= "popMode()"

LexerTypeCommand(arg, grammar) ::= "_type = <arg>"

LexerChannelCommand(arg, grammar) ::= "_channel = <arg>"

LexerModeCommand(arg, grammar) ::= "_mode = <arg>"

LexerPushModeCommand(arg, grammar) ::= "pushMode(<arg>)"

ActionText(t) ::= "<t.text>"

ActionTemplate(t) ::= "<t.st>"

ArgRef(a) ::= "localctx.<a.name>"

LocalRef(a) ::= "localctx.<a.name>"

RetValRef(a) ::= "localctx.<a.name>"

QRetValRef(a) ::= "<ctx(a)>.<a.dict>.<a.name>"

/** How to translate \$tokenLabel */

TokenRef(t) ::= "<ctx(t)>.<t.name>"

LabelRef(t) ::= "<ctx(t)>.<t.name>"

ListLabelRef(t) ::= "<ctx(t)>.<ListLabelName(t.name)>"

SetAttr(s,rhsChunks) ::= "<ctx(s)>.<s.name> = <rhsChunks>"

TokenLabelType() ::= "<file.TokenLabelType; null={Token}>"

InputSymbolType() ::= "<file.InputSymbolType; null={Token}>"

TokenPropertyRef_text(t) ::= "(None if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.text)"

TokenPropertyRef_type(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.type)"

TokenPropertyRef_line(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.line)"

TokenPropertyRef_pos(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.column)"

TokenPropertyRef_channel(t) ::= "(0 if (<ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.channel)"

TokenPropertyRef_index(t) ::= "(0 if <ctx(t)>.<t.label> is None else <ctx(t)>.<t.label>.tokenIndex)"

TokenPropertyRef_int(t) ::= "(0 if <ctx(t)>.<t.label> is None else int(<ctx(t)>.<t.label>.text))"

RulePropertyRef_start(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.start)"

RulePropertyRef_stop(r) ::= "(None if <ctx(r)>.<r.label> is None else <ctx(r)>.<r.label>.stop)"

RulePropertyRef_text(r) ::= "(None if <ctx(r)>.<r.label> is None else
self._input.getText((<ctx(r)>.<r.label>.start,<ctx(r)>.<r.label>.stop)))"

RulePropertyRef_ctx(r) ::= "<ctx(r)>.<r.label>"

RulePropertyRef_parser(r) ::= "self"

ThisRulePropertyRef_start(r) ::= "localctx.start"

ThisRulePropertyRef_stop(r) ::= "localctx.stop"

ThisRulePropertyRef_text(r) ::= "self._input.getText((localctx.start, self._input.LT(-1)))"

ThisRulePropertyRef_ctx(r) ::= "localctx"

ThisRulePropertyRef_parser(r) ::= "self"

NonLocalAttrRef(s) ::= "getInvokingContext(<s.ruleIndex>).<s.name>"

```
SetNonLocalAttr(s, rhsChunks) ::= "getInvokingContext(<s.ruleIndex>).<s.name> = <rhsChunks>"
```

```
AddToLabelList(a) ::= "<ctx(a.label)>.<a.listName>.append(<labelref(a.label)>)"
```

```
TokenDecl(t) ::= "self.<t.name> = None # <TokenLabelType()>"
```

```
TokenTypeDecl(t) ::= "self.<t.name> = 0 # <TokenLabelType()> type"
```

```
TokenListDecl(t) ::= "self.<t.name> = list() # of <TokenLabelType()>s"
```

```
RuleContextDecl(r) ::= "self.<r.name> = None # <r.ctxName>"
```

```
RuleContextListDecl(rdecl) ::= "self.<rdecl.name> = list() # of <rdecl.ctxName>s"
```

```
ContextTokenGetterDecl(t) ::= <<
```

```
def <t.name>(self):
```

```
    return self.getToken(<parser.name>.<t.name>, 0)
```

```
>>
```

```
// should never be called
```

```
ContextTokenListGetterDecl(t) ::= <<
```

```
def <t.name>_list(self):
```

```
    return self.getTokens(<parser.name>.<t.name>)
```

```
>>
```

```
ContextTokenListIndexedGetterDecl(t) ::= <<
```

```
def <t.name>(self, i:int=None):
```

```
    if i is None:
```

```
        return self.getTokens(<parser.name>.<t.name>)
```

```
    else:
```

```
        return self.getToken(<parser.name>.<t.name>, i)
```

```
>>
```

```
ContextRuleGetterDecl(r) ::= <<
```

```
def <r.name>(self):
```

```
    return self.getTypedRuleContext(<parser.name>.<r.ctxName>,0)
```

```
>>
```

```
// should never be called
```

```
ContextRuleListGetterDecl(r) ::= <<
```

```
def <r.name>_list(self):
```

```
    return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
```

```
>>
```

```
ContextRuleListIndexedGetterDecl(r) ::= <<
```

```
def <r.name>(self, i:int=None):
```

```
    if i is None:
```

```
        return self.getTypedRuleContexts(<parser.name>.<r.ctxName>)
```

```
    else:
```

```
        return self.getTypedRuleContext(<parser.name>.<r.ctxName>,i)
```

>>

```
LexerRuleContext() ::= "RuleContext"
```

```
/** The rule context name is the rule followed by a suffix; e.g.,  
 * r becomes rContext.  
 */
```

```
RuleContextNameSuffix() ::= "Context"
```

```
ImplicitTokenLabel(tokenName) ::= "_<tokenName>"
```

```
ImplicitRuleLabel(ruleName) ::= "_<ruleName>"
```

```
ImplicitSetLabel(id) ::= "_tset<id>"
```

```
ListLabelName(label) ::= "<label>"
```

```
CaptureNextToken(d) ::= "<d.varName> = self._input.LT(1)"
```

```
CaptureNextTokenType(d) ::= "<d.varName> = self._input.LA(1)"
```

```
StructDecl(struct,ctorAttrs,attrs, getters,dispatchMethods,interfaces,extensionMembers) ::= <<  
class <struct.name>(<if(contextSuperClass)><contextSuperClass><else>ParserRuleContext<endif>):
```

```
    def __init__(self, parser, parent:ParserRuleContext=None, invokingState:int=-1<struct.ctorAttrs:{ a | ,  
<a.name><if(a.type)>:<a.type><endif>=None }):
```

```
        super().__init__(parent, invokingState)
```

```
        self.parser = parser
```

```
        <attrs:{ a | <a> }; separator="\n">
```

```
        <struct.ctorAttrs:{ a | self.<a.name> = <a.name> }; separator="\n">
```

```
    <getters:{ g | <g> }; separator="\n\n">
```

```
    def getRuleIndex(self):
```

```
        return <parser.name>.RULE_<struct.derivedFromName>
```

```
<if(struct.provideCopyFrom)> <! don't need copy unless we have subclasses !>
```

```
    def copyFrom(self, ctx:ParserRuleContext):
```

```
        super().copyFrom(ctx)
```

```
        <struct.attrs:{ a | self.<a.name> = ctx.<a.name> }; separator="\n">
```

```
<endif>
```

```
    <dispatchMethods; separator="\n">
```

```
    <extensionMembers; separator="\n">
```

>>

```
AltLabelStructDecl(struct,attrs, getters,dispatchMethods) ::= <<
```

```
class <struct.name>(<currentRule.name; format="cap">Context):
```

```
    def __init__(self, parser, ctx:ParserRuleContext): # actually a <parser.name>.<currentRule.name;
```

```

format="cap">Context
  super().__init__(parser)
  <attrs:{ a | <a>}; separator="\n">
  self.copyFrom(ctx)

  <getters:{ g | <g>}; separator="\n">

  <dispatchMethods; separator="\n">

>>

ListenerDispatchMethod(method) ::= <<
def <if(method.isEnter)>enter<else>exit<endif>Rule(self, listener:ParseTreeListener):
  if hasattr( listener, "<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">" ):
    listener.<if(method.isEnter)>enter<else>exit<endif><struct.derivedFromName; format="cap">(self)

>>

VisitorDispatchMethod(method) ::= <<
def accept(self, visitor:ParseTreeVisitor):
  if hasattr( visitor, "visit<struct.derivedFromName; format="cap">" ):
    return visitor.visit<struct.derivedFromName; format="cap">(self)
  else:
    return visitor.visitChildren(self)

>>

AttributeDecl(d) ::= "self.<d.name> = <if(d.initValue)><d.initValue><else>None<endif>"

/** If we don't know location of label def x, use this template */
labelref(x) ::= "<if(!x.isLocal)>localctx.<endif><x.name>"

/** For any action chunk, what is correctly-typed context struct ptr? */
ctx(actionChunk) ::= "localctx"

// used for left-recursive rules
recRuleAltPredicate(ruleName,opPrec) ::= "self.precpred(self._ctx, <opPrec>)"
recRuleSetReturnAction(src,name) ::= "$<name>=<src>.<name>"
recRuleSetStopToken() ::= "self._ctx.stop = self._input.LT(-1)"

recRuleAltStartAction(ruleName, ctxName, label) ::= <<
localctx = <parser.name>.<ctxName>Context(self, _parentctx, _parentState)
<if(label)>localctx.<label> = _prevctx<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

recRuleLabeledAltStartAction(ruleName, currentAltLabel, label, isListLabel) ::= <<
localctx = <parser.name>.<currentAltLabel; format="cap">Context(self, <parser.name>.<ruleName>

```

```

format="cap">Context(self, _parentctx, _parentState))
<if(label)>
<if(isListLabel)>
localctx.<label>.append(_prevctx)
<else>
localctx.<label> = _prevctx
<endif>
<endif>
self.pushNewRecursionContext(localctx, _startState, self.RULE_<ruleName>)
>>

```

```

recRuleReplaceContext(ctxName) ::= <<
localctx = <parser.name>.<ctxName>Context(self, localctx)
self._ctx = localctx
_prevctx = localctx
>>

```

```

recRuleSetPrevCtx() ::= <<
if self._parseListeners is not None:
    self.triggerExitRuleEvent()
_prevctx = localctx
>>

```

```

LexerFile(lexerFile, lexer, namedActions) ::= <<
<fileHeader(lexerFile.grammarFileName, lexerFile.ANTLRVersion)>
from antlr4 import *
from io import StringIO
from typing.io import TextIO
import sys

<namedActions.header>

<lexer>
>>

```

```

Lexer(lexer, atn, actionFuncs, sempredFuncs, superClass) ::= <<
<if(superClass)>
if __name__ is not None and "." in __name__:
    from .<superClass> import <superClass>
else:
    from <superClass> import <superClass>

<endif>
<atn>

class <lexer.name>(<if(superClass)><superClass><else>Lexer<endif>):

```

```

atn = ATNDeserializer().deserialize(serializedATN())

decisionsToDFA = [ DFA(ds, i) for i, ds in enumerate(atn.decisionToState) ]

<if(lexer.channels)>
  <lexer.channels:{c| <c> = <lexer.channels.(c)>}; separator="\n">

<endif>
<if(rest(lexer.modes))>
  <rest(lexer.modes):{m| <m> = <i>}; separator="\n">

<endif>
  <lexer.tokens:{k | <k> = <lexer.tokens.(k)>}; separator="\n", wrap, anchor>

channelNames = [ u"DEFAULT_TOKEN_CHANNEL", u"HIDDEN"<if (lexer.channels)>, <lexer.channels:{c|
u"<c>"}; separator=", ", wrap, anchor><endif> ]

modeNames = [ <lexer.modes:{m| "<m>"}; separator=", ", wrap, anchor> ]

literalNames = [ "\<INVALID>",
  <lexer.literalNames:{t | <t>}; separator=", ", wrap, anchor> ]

symbolicNames = [ "\<INVALID>",
  <lexer.symbolicNames:{t | <t>}; separator=", ", wrap, anchor> ]

ruleNames = [ <lexer.ruleNames:{r | "<r>"}; separator=", ", wrap, anchor> ]

grammarFileName = "<lexer.grammarFileName>"

def __init__(self, input=None, output:TextIO = sys.stdout):
  super().__init__(input, output)
  self.checkVersion("<lexerFile.ANTLRVersion>")
  self._interp = LexerATNSimulator(self, self.atn, self.decisionsToDFA, PredictionContextCache())
  self._actions = None
  self._predicates = None

<namedActions.members>

<dumpActions(lexer, "", actionFuncs, sempredFuncs)>

>>

SerializedATN(model) ::= <<
<! only one segment, can be inlined !>

def serializedATN():
  with StringIO() as buf:
    buf.write("<model.serialized; wrap={ }<\n>    buf.write(">")>")

```



```
return buf.getvalue()
```

```
>>
```

```
/** Using a type to init value map, try to init a type; if not in table
```

```
* must be an object, default value is "null".
```

```
*/
```

```
initValue(typeName) ::= <<
```

```
<pythonTypeInitMap.(typeName)>
```

```
>>
```

```
codeFileExtension() ::= ".py"
```

Found in path(s):

```
* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
```

```
jar/org/antlr/v4/tool/templates/codegen/Python3/Python3.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* Copyright (c) 2012-2017 The ANTLR Project. All rights reserved.
```

```
* Use of this file is governed by the BSD 3-clause license that
```

```
* can be found in the LICENSE.txt file in the project root.
```

```
*/
```

```
/**
```

```
[The "BSD license"]
```

```
Copyright (c) 2011 Cay Horstmann
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/gui/GraphicsSupport.java

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012-2016 Terence Parr

* Copyright (c) 2012-2016 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
tree grammar SourceGenTriggers;
```

```
options {
```

```
language = Java;
```

```
tokenVocab = ANTLRParser;
```

```
ASTLabelType = GrammarAST;
```

```
}
```

```
@header {
```

```
package org.antlr.v4.codegen;
```

```
import org.antlr.v4.misc.Utils;
```

```

import org.antlr.v4.codegen.model.*;
import org.antlr.v4.codegen.model.decl.*;
import org.antlr.v4.tool.*;
import org.antlr.v4.tool.ast.*;
import java.util.Collections;
import java.util.Map;
import java.util.HashMap;
}

```

```

@members {
public OutputModelController controller;
public boolean hasLookaheadBlock;
public SourceGenTriggers(TreeNodeStream input, OutputModelController controller) {
this(input);
this.controller = controller;
}
}

```

```

dummy : block[null, null] ;

```

```

block[GrammarAST label, GrammarAST ebnfRoot] returns [List<? extends SrcOp> omos]
: ^( blk=BLOCK (^ (OPTIONS .+))?
{List<CodeBlockForAlt> alts = new ArrayList<CodeBlockForAlt>();
( alternative {alts.add($alternative.altCodeBlock);} )+
)
{
if ( alts.size()==1 && ebnfRoot==null) return alts;
if ( ebnfRoot==null ) {
Somos = DefaultOutputModelFactory.list(controller.getChoiceBlock((BlockAST)$blk, alts, $label));
}
else {
Choice choice = controller.getEBNFBlock($ebnfRoot, alts);
hasLookaheadBlock |= choice instanceof PlusBlock || choice instanceof StarBlock;
Somos = DefaultOutputModelFactory.list(choice);
}
}
;

```

```

alternative returns [CodeBlockForAlt altCodeBlock, List<SrcOp> ops]

```

```

@init {
boolean outerMost = inContext("RULE BLOCK");
}
@after {
controller.finishAlternative($altCodeBlock, $ops, outerMost);
}
: a=alt[outerMost] { $altCodeBlock=$a.altCodeBlock; $ops=$a.ops; }
;

```

```

alt[boolean outerMost] returns [CodeBlockForAlt altCodeBlock, List<SrcOp> ops]
@init {
// set alt if outer ALT only (the only ones with alt field set to Alternative object)
AltAST altAST = (AltAST)retval.start;
if ( outerMost ) controller.setCurrentOuterMostAlt(altAST.alt);
}
: {
List<SrcOp> elems = new ArrayList<SrcOp>();
// TODO: shouldn't we pass $start to controller.alternative()?
$altCodeBlock = controller.alternative(controller.getCurrentOuterMostAlt(), outerMost);
$altCodeBlock.ops = $ops = elems;
controller.setCurrentBlock($altCodeBlock);
}
^( ALT elementOptions? ( element {if ($element.omos!=null) elems.addAll($element.omos);} )+ )

| ^(ALT elementOptions? EPSILON)
    { $altCodeBlock = controller.epsilon(controller.getCurrentOuterMostAlt(), outerMost); }
;

element returns [List<? extends SrcOp> omos]
: labeledElement { $omos = $labeledElement.omos; }
| atom[null,false] { $omos = $atom.omos; }
| subrule { $omos = $subrule.omos; }
| ACTION { $omos = controller.action((ActionAST)$ACTION); }
| SEMPRED { $omos = controller.sempred((ActionAST)$SEMPRED); }
| ^(ACTION elementOptions) { $omos = controller.action((ActionAST)$ACTION); }
| ^(SEMPRED elementOptions) { $omos = controller.sempred((ActionAST)$SEMPRED); }
;

labeledElement returns [List<? extends SrcOp> omos]
: ^(ASSIGN ID atom[$ID,false] ) { $omos = $atom.omos; }
| ^(PLUS_ASSIGN ID atom[$ID,false]) { $omos = $atom.omos; }
| ^(ASSIGN ID block[$ID,null] ) { $omos = $block.omos; }
| ^(PLUS_ASSIGN ID block[$ID,null]) { $omos = $block.omos; }
;

subrule returns [List<? extends SrcOp> omos]
: ^(OPTIONAL b=block[null,$OPTIONAL])
{
$omos = $block.omos;
}
| ( ^(op=CLOSURE b=block[null,null])
| ^(op=POSITIVE_CLOSURE b=block[null,null])
)
{
List<CodeBlockForAlt> alts = new ArrayList<CodeBlockForAlt>();
SrcOp blk = $b.omos.get(0);
CodeBlockForAlt alt = new CodeBlockForAlt(controller.delegate);

```

```

alt.addOp(blk);
alts.add(alt);
SrcOp loop = controller.getEBNFBlock($op, alts); // "star it"
    hasLookaheadBlock |= loop instanceof PlusBlock || loop instanceof StarBlock;
    $omos = DefaultOutputModelFactory.list(loop);
}
| block[null, null]    {$omos = $block.omos;}
;

blockSet[GrammarAST label, boolean invert] returns [List<SrcOp> omos]
: ^(SET atom[label,invert]+) {$omos = controller.set($SET, $label, invert);}
;

/*
setElement
: STRING_LITERAL
| TOKEN_REF
| ^(RANGE STRING_LITERAL STRING_LITERAL)
;
*/

// TODO: combine ROOT/BANG into one then just make new op ref'ing return value of atom/terminal...
// TODO: same for NOT
atom[GrammarAST label, boolean invert] returns [List<SrcOp> omos]
: ^(NOT a=atom[$label, true]) {$omos = $a.omos;}
| range[label]    {$omos = $range.omos;}
| ^(DOT ID terminal[$label])
| ^(DOT ID ruleref[$label])
| ^(WILDCARD .)    {$omos = controller.wildcard($WILDCARD, $label);}
| WILDCARD        {$omos = controller.wildcard($WILDCARD, $label);}
| terminal[label]  {$omos = $terminal.omos;}
| ruleref[label]   {$omos = $ruleref.omos;}
| blockSet[$label, invert] {$omos = $blockSet.omos;}
;

ruleref[GrammarAST label] returns [List<SrcOp> omos]
: ^(RULE_REF ARG_ACTION? elementOptions?) {$omos = controller.ruleRef($RULE_REF, $label,
$ARG_ACTION);}
;

range[GrammarAST label] returns [List<SrcOp> omos]
: ^(RANGE a=STRING_LITERAL b=STRING_LITERAL)
;

terminal[GrammarAST label] returns [List<SrcOp> omos]
: ^(STRING_LITERAL .) {$omos = controller.stringRef($STRING_LITERAL, $label);}
| STRING_LITERAL    {$omos = controller.stringRef($STRING_LITERAL, $label);}
| ^(TOKEN_REF ARG_ACTION .) {$omos = controller.tokenRef($TOKEN_REF, $label, $ARG_ACTION);}

```

```
|^(TOKEN_REF .) {$omos = controller.tokenRef($TOKEN_REF, $label, null);}
| TOKEN_REF {$omos = controller.tokenRef($TOKEN_REF, $label, null);}
;
```

elementOptions

```
:^(ELEMENT_OPTIONS elementOption+)
;
```

elementOption

```
: ID
| ^(ASSIGN ID ID)
| ^(ASSIGN ID STRING_LITERAL)
| ^(ASSIGN ID ACTION)
| ^(ASSIGN ID INT)
;
```

Found in path(s):

```
*/opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-
jar/org/antlr/v4/codegen/SourceGenTriggers.g
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD licence"]
```

```
Copyright (c) 2005-2009 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-jar/org/antlr/v4/parse/ANTLRLexer.java

No license file was found, but licenses were detected in source scan.

/*

[The "BSD licence"]

Copyright (c) 2006 Kay Roepke

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

This file contains the actual layout of the messages emitted by ANTLR.

This file contains the default format ANTLR uses.

*/

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<id> <text>"

report(location, message, type) ::= "<type><message.id>: <location> <message.text>"

wantsSingleLineMessage() ::= "false"

Found in path(s):

* /opt/cola/permits/1192109359_1629132978.03/0/antlr4-4-7-2-sources-1-

jar/org/antlr/v4/tool/templates/messages/formats/antlr.stg

1.187 open-telemetry/opentelemetry-java

1.6.0

1.187.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// Copyright 2008 Google Inc. All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions are
// * Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// * Redistributions in binary form must reproduce the above
// copyright notice, this list of conditions and the following disclaimer
// in the documentation and/or other materials provided with the
// * Neither the name of Google Inc. nor the names of its
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1310721497_1650631670.66/0/opentelemetry-exporter-otlp-common-1-6-0-sources-
jar/io/opentelemetry/exporter/otlp/internal/CodedOutputStream.java
* /opt/cola/permits/1310721497_1650631670.66/0/opentelemetry-exporter-otlp-common-1-6-0-sources-
jar/io/opentelemetry/exporter/otlp/internal/WireFormat.java
* /opt/cola/permits/1310721497_1650631670.66/0/opentelemetry-exporter-otlp-common-1-6-0-sources-
jar/io/opentelemetry/exporter/otlp/internal/Utf8.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 The gRPC Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1310721497_1650631670.66/0/opentelemetry-exporter-otlp-common-1-6-0-sources-
jar/io/opentelemetry/exporter/otlp/internal/grpc/MarshalerInputStream.java
```


1.188 micronaut-kubernetes 3.4.0

1.188.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2021 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/PodNameResolver.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/NamespaceResolver.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/Apis.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/OkHttpClientLogging.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/ApisAutomaticFeatureMetadata.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/ApiClientConfiguration.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/DefaultNamespaceResolver.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/ApiClientFactory.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/DiscoveryFactory.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/graalvm/KubernetesClientFeature.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/ModelMapper.java
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-
jar/io/micronaut/kubernetes/client/DefaultPodNameResolver.java
```

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2022 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331474306_1653514752.2550623/0/micronaut-kubernetes-client-3-4-0-sources-  
jar/io/micronaut/kubernetes/client/DiscoveryCache.java
```

1.189 java-servlet-api 4.0.4

1.189.1 Available under license :

Found license 'General Public License 2.0' in 'Copyright (c) 1997, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License, Copyright © 2019 Eclipse Foundation. All rights reserved.
'

Found license 'General Public License 2.0' in 'Copyright (c) 2008, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'General Public License 2.0' in 'Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 1997, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License, Copyright © 2019 Eclipse Foundation. All rights reserved.
'

Found license 'Eclipse Public License 1.0' in 'Copyright (c) 2008, 2018 Oracle and/or its affiliates. All rights reserved. This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0, which is available at Eclipse Public License v. 2.0 are satisfied: GNU General Public License,'

1.190 osgi-resource-locator-bundle 1.0.3

1.190.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (c) 2009, 2018 Oracle and/or its affiliates. All rights reserved.  
*  
* This program and the accompanying materials are made available under the  
* terms of the Eclipse Public License v. 2.0, which is available at  
* http://www.eclipse.org/legal/epl-2.0.  
*  
* This Source Code may also be made available under the following Secondary  
* Licenses when the conditions for such availability set forth in the  
* Eclipse Public License v. 2.0 are satisfied: GNU General Public License,  
* version 2 with the GNU Classpath Exception, which is available at  
* https://www.gnu.org/software/classpath/license.html.  
*  
* SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0  
*/
```

Found in path(s):

```
* /opt/cola/permits/1340031775_1654680102.675777/0/osgi-resource-locator-1-0-3-sources-2-  
jar/org/glassfish/hk2/osgiresourcelocator/ServiceLoader.java  
* /opt/cola/permits/1340031775_1654680102.675777/0/osgi-resource-locator-1-0-3-sources-2-  
jar/org/glassfish/hk2/osgiresourcelocator/ResourceFinder.java  
* /opt/cola/permits/1340031775_1654680102.675777/0/osgi-resource-locator-1-0-3-sources-2-  
jar/org/glassfish/hk2/osgiresourcelocator/Activator.java  
* /opt/cola/permits/1340031775_1654680102.675777/0/osgi-resource-locator-1-0-3-sources-2-  
jar/org/glassfish/hk2/osgiresourcelocator/ServiceLoaderImpl.java  
* /opt/cola/permits/1340031775_1654680102.675777/0/osgi-resource-locator-1-0-3-sources-2-  
jar/org/glassfish/hk2/osgiresourcelocator/ResourceFinderImpl.java
```

1.191 ktor-network 1.2.6

1.191.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.192 io.grpc:grpc-bom 1.39.0

1.192.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications

represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without

modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade

names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier

identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise

designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must

include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly

negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified portion of 'OkHttp', an open source
HTTP & SPDY client for Android and Java applications, which can be obtained
at:

* LICENSE:
* okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
* HOMEPAGE:
* <https://github.com/square/okhttp>
* LOCATION_IN_GRPC:
* okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source
cloud-native high-performance edge/middle/service proxy, which can be
obtained at:

* LICENSE:
* xds/third_party/envoy/LICENSE (Apache License 2.0)
* NOTICE:
* xds/third_party/envoy/NOTICE

- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
 - * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa', an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * xds/third_party/udpa/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.193 micronaut 3.4.1

1.193.1 Available under license :

Micronaut includes references to parts of GraalVM 21.0 or later distributed under the UPL license below:

Copyright (c) 2018 - 2021 Oracle and/or its affiliates. All rights reserved

The Universal Permissive License (UPL), Version 1.0

Subject to the condition set forth below, permission is hereby granted to any person obtaining a copy of this software, associated documentation and/or data (collectively the "Software"), free of charge and under any and all copyright rights in the Software, and any and all patent rights owned or freely licensable by each licensor hereunder covering either (i) the unmodified Software as contributed to or provided by such licensor, or (ii) the Larger Works (as defined below), to deal in both

(a) the Software, and

(b) any piece of software and/or hardware listed in the lgrwrks.txt file if one is included with the Software (each a Larger Work to which the Software is contributed by such licensors),

without restriction, including without limitation the rights to copy, create derivative works of, display, perform, and distribute the Software and make, use, sell, offer for sale, import, export, have made, and have sold the Software and the Larger Work(s), and to sublicense the foregoing rights on either these or other terms.

This license is subject to the following condition:

The above copyright notice and either this complete permission notice or at a minimum a reference to the UPL must be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License

Version 2.0, January 2004

<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.194 apache-http-client 4.5.13

1.194.1 Available under license :

Apache HttpComponents Client
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

=====

This project includes Public Suffix List copied from
<https://publicsuffix.org/list/effective_tld_names.dat>
licensed under the terms of the Mozilla Public License, v. 2.0

Full license text: <<http://mozilla.org/MPL/2.0/>>

Mozilla Public License Version 2.0

=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution

become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License.

Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software;
or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted

in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of

the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions,

* limitation may not apply to You. *
* *

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <http://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

1.195 byte-buddy-agent 1.12.7

1.195.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright \${project.inceptionYear} - \${current.year} \${copyright.holder}

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.196 fabric8-::-kubernetes-model-::-core

4.10.3

1.196.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by

the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained

within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.197 google-guice 4.1.0

1.197.1 Available under license :

Google Guice - Core Library
Copyright 2006-2016 Google, Inc.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.198 kotlinox-coroutines-core 1.3.2-1.3.60

1.198.1 Available under license :

/*

* Copyright 2016-2018 JetBrains s.r.o. Use of this source code is governed by the Apache 2.0 license.

*/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or

otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual,

worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents

of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.199 fabric8-::-kubernetes-model-::-settings

4.10.3

1.199.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.200 jackson-dataformat-yaml 2.13.2

1.200.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.201 jackson-databind 2.13.2.2

1.201.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has

been in development since 2007.

It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0

To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.202 micronaut-gcp 4.1.1

1.202.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2020 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
```

jar/io/micronaut/gcp/GoogleCloudConfiguration.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/GoogleComputeMetadataKeys.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/credentials/GoogleCredentialsConfiguration.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/GoogleComputeInstanceMetadataResolver.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/UserAgentHeaderProvider.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/condition/RequiresProjectIdCondition.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/package-info.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/Modules.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/GoogleComputeNetworkInterface.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/GoogleComputeInstanceMetadata.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/discovery/cloud/gcp/GoogleComputeMetadataConfiguration.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/credentials/GoogleCredentialsFactory.java
* /opt/cola/permits/1331473977_1653513089.549853/0/micronaut-gcp-common-4-1-1-sources-
jar/io/micronaut/gcp/condition/RequiresGoogleProjectId.java

1.203 classgraph 4.8.25

1.203.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* This file is part of ClassGraph.  
*  
* Author: R. Kempees  
*  
* Hosted at: https://github.com/classgraph/classgraph  
*  
* --  
*  
* The MIT License (MIT)  
*  
* Copyright (c) 2017 R. Kempees  
*  
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated  
* documentation files (the "Software"), to deal in the Software without restriction, including without  
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
```

* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/WebSphereLibertyClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Michael J. Simons

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2019 Luke Hutchison

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:

*

* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT

* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO

* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/SpringBootRestartClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Harith Elrufaie

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2017 Harith Elrufaie

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT

* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO

* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN

* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE

* OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/FelixClassLoaderHandler.java

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: Luke Hutchison (luke.hutch@gmail.com)
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
 *
 * The MIT License (MIT)
 *
 * Copyright (c) 2019 Luke Hutchison
 *
 * Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
 * documentation files (the "Software"), to deal in the Software without restriction, including without
 * limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
 * the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
 * conditions:
 *
 * The above copyright notice and this permission notice shall be included in all copies or substantial
 * portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT
 * LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
 * AND NONINFRINGEMENT. IN NO
 * EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
 * OR OTHER LIABILITY, WHETHER IN
 * AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
 * WITH THE SOFTWARE OR THE USE
 * OR OTHER DEALINGS IN THE SOFTWARE.
 */
```

Found in path(s):

```
*/opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/ResourceList.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * This file is part of ClassGraph.
 *
 * Author: Sergey Bespalov
 *
 * Hosted at: https://github.com/classgraph/classgraph
 *
 * --
```

*
* The MIT License (MIT)
*
* Copyright (c) 2017 Sergey Bepalov
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
* OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/WebSphereTraditionalClassLoaderHandler.java
No license file was found, but licenses were detected in source scan.

/*
* This file is part of ClassGraph.
*
* Author: Luke Hutchison
*
* Hosted at: <https://github.com/classgraph/classgraph>
*
* --
*
* The MIT License (MIT)
*
* Copyright (c) 2019 Luke Hutchison
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:

*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/Classfile.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/ModuleInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/MethodInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/concurrency/InterruptedException.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/json/JSONParser.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxContextFinderClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/recycler/Resetable.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/AnnotationClassRef.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/fastzipfilereader/RecyclableInflater.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/ClasspathElementDir.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/WeblogicClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/ObjectTypedValueWrapper.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/io/github/classgraph/ClassGraph.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/json/FieldTypeInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/TomcatWebappClassLoaderBaseHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/fastzipfilereader/LogicalZipFile.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/types/Parser.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ModulePathInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/types/TypeUtils.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/fastzipfilereader/PhysicalZipFile.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/SystemJarFinder.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ReferenceTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONObject.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/PackageInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/ClassLoaderHandlerRegistry.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/JarUtils.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/CallStackReader.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/TypeArgument.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONArray.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/HierarchicalTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/ClassFieldCache.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/URLPathEncoder.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/fastzipfilereader/ZipFileSliceReader.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/VersionFinder.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/MethodParameterInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/InfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/PackageInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ModuleInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/ReflectionUtils.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/ClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/fastzipfilereader/ZipFileSlice.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/FieldInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/utils/LogNode.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/AnnotationEnumValue.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ScanResultObject.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/URLClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONUtils.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ModuleReaderProxy.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/MethodTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/FallbackClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/ClassLoaderOrder.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/AnnotationInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/AnnotationParameterValueList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClasspathElementZip.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/WhiteBlackList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/ClassLoaderAndModuleFinder.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/utils/InputStreamOrByteBufferAdapter.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/TypeParameter.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassRefTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/ReferenceEqualityKey.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ArrayTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/MethodInfoList.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/concurrency/SingletonMap.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/ParameterizedTypeImpl.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/concurrency/WorkQueue.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/PlexusClassWorldsClassRealmClassLoaderHandler.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/ClassFields.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/types/ParseException.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONReference.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/Resource.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/fastzipfilereader/NestedJarHandler.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/recycler/Recycler.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/Join.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/TypeVariableSignature.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassRefOrTypeVariableSignature.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/ClasspathFinder.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassInfo.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/EquinoxClassLoaderHandler.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ScanResult.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classpath/ClasspathOrder.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClasspathElementModule.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/fastzipfilereader/FastZipEntry.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/AnnotationParameterValue.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassGraphClassLoader.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/GraphvizDotfileGenerator.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/ScanSpec.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONSerializer.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/ParentLastDelegationOrderTestClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/MappableInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/Scanner.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/BaseTypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/OSGiDefaultClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/AnnotationInfoList.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/CollectionUtils.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/classloaderhandler/JPMSCClassLoaderHandler.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/recycler/RecycleOnClose.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/FastPathResolver.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/Utils/FileUtils.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/TypeResolutions.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/HasName.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/FieldInfo.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/TypeSignature.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClassGraphException.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ModuleRef.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/Id.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/io/github/classgraph/ClasspathElement.java
* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/nonapi/io/github/classgraph/json/JSONDeserializer.java
No license file was found, but licenses were detected in source scan.

<name>The MIT License (MIT)</name>

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-jar/META-

INF/maven/io.github.classgraph/classgraph/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: @jacobg on GitHub

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2019 @jacobg, Luke Hutchison

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT

* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO

* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN

* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE

* OR OTHER DEALINGS IN THE SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-

jar/nonapi/io/github/classgraph/classloaderhandler/AntClassLoaderHandler.java

No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Johnno Crawford (johnno@sulake.com)

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --
*
* The MIT License (MIT)
*
* Copyright (c) 2016 Johno Crawford
*
* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated
* documentation files (the "Software"), to deal in the Software without restriction, including without
* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
* IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
* AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
* OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
* WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/concurrency/SimpleThreadFactory.java

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/concurrency/AutoCloseableExecutorService.java

No license file was found, but licenses were detected in source scan.

/*

* This file is part of ClassGraph.

*

* Author: Luke Hutchison

*

* Hosted at: <https://github.com/classgraph/classgraph>

*

* --

*

* The MIT License (MIT)

*

* Copyright (c) 2019 Luke Hutchison, with significant contributions from Davy De Durpel

*

* Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated

* documentation files (the "Software"), to deal in the Software without restriction, including without

* limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of
* the Software, and to permit persons to whom the Software is furnished to do so, subject to the following
* conditions:
*
* The above copyright notice and this permission notice shall be included in all copies or substantial
* portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT
* LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE
AND NONINFRINGEMENT. IN NO
* EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES
OR OTHER LIABILITY, WHETHER IN
* AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION
WITH THE SOFTWARE OR THE USE
* OR OTHER DEALINGS IN THE SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1162864585_1620923804.39/0/classgraph-4-8-25-sources-1-
jar/nonapi/io/github/classgraph/classloaderhandler/JBossClassLoaderHandler.java

1.204 arrow-core-data 0.9.0

1.204.1 Available under license :

Apache-2.0

1.205 bom 4.1.74.Final

1.205.1 Available under license :

Copyright (c) 2003-2008 Yuta Mori All Rights Reserved.

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the
Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES

OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuity of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common

control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or

documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill,

work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but

not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their

Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with

the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2009 William Kinney

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING

FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Copyright 2009-2010 Ning, Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright (c) 2004-2006, 2008, 2009, 2011 Apple Inc. All rights reserved.

*
* @APPLE_LICENSE_HEADER_START@
*
* This file contains Original Code and/or Modifications of Original Code
* as defined in and that are subject to the Apple Public Source License
* Version 2.0 (the 'License'). You may not use this file except in
* compliance with the License. Please obtain a copy of the License at
* <https://www.opensource.apple.com/apsl/> and read it before using this
* file.
*
* The Original Code and all software distributed under the License are
* distributed on an 'AS IS' basis, WITHOUT WARRANTY OF ANY KIND, EITHER
* EXPRESS OR IMPLIED, AND APPLE HEREBY DISCLAIMS ALL SUCH WARRANTIES,
* INCLUDING WITHOUT LIMITATION, ANY WARRANTIES OF MERCHANTABILITY,
* FITNESS FOR A PARTICULAR PURPOSE, QUIET ENJOYMENT OR NON-INFRINGEMENT.
* Please see the License for the specific language governing rights and
* limitations under the License.
*
* @APPLE_LICENSE_HEADER_END@
*/

The MIT License (MIT)

Copyright (c) 2014 Cory Benfield

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 2011, Google Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
Copyright (c) 2010-2011 Matthew J. Francis and Contributors of the jbzip2 Project

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The person or persons who have associated work with this document (the "Dedicator" or "Certifier") hereby either (a) certifies that, to the best of his knowledge, the work of authorship identified is in the public domain of the country from which the work is published, or (b) hereby dedicates whatever copyright the dedicators holds in the work of authorship identified below (the "Work") to the public domain. A certifier, moreover, dedicates any copyright interest he may have in the associated work, and for these purposes, is described as a "dedicator" below.

A certifier has taken reasonable steps to verify the copyright status of this work. Certifier recognizes that his good faith efforts may not shield him from liability if in fact the work certified is not in the public domain.

Dedicator makes this dedication for the benefit of the public at large and to the detriment of the Dedicator's heirs and successors. Dedicator intends this dedication to be an overt act of relinquishment in perpetuate of all present and future rights under copyright law, whether vested or contingent, in the Work. Dedicator understands that such relinquishment of all rights includes the relinquishment of all rights to enforce (by lawsuit or otherwise) those copyrights in the Work.

Dedicator recognizes that, once placed in the public domain, the Work may be freely reproduced, distributed, transmitted, used, modified, built upon, or otherwise exploited by anyone for any purpose, commercial or non-commercial, and in any way, including by methods that have not yet been invented or conceived.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (c) 2012, 2014, 2015, 2016 Tatsuhiro Tsujikawa
Copyright (c) 2012, 2014, 2015, 2016 nghttp2 contributors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION

OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

(BSD License: <https://www.opensource.org/licenses/bsd-license>)

Copyright (c) 2011, Joe Walnes, Aslak Hellesøy and contributors
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the
following conditions are met:

- * Redistributions of source code must retain the above
copyright notice, this list of conditions and the
following disclaimer.
- * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the
following disclaimer in the documentation and/or other
materials provided with the distribution.
- * Neither the name of the Webbit nor the names of
its contributors may be used to endorse or promote products
derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND
CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES,
INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF
MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES
(INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE
GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE
POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2000 - 2013 The Legion of the Bouncy Castle Inc.
(<https://www.bouncycastle.org>)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (c) 2000,2001,2002,2003,2004 ymnk, JCraft,Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The Netty Project

=====

Please visit the Netty web site for more information:

* <https://netty.io/>

Copyright 2014 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Also, please refer to each LICENSE.<component>.txt file, which is located in the 'license' directory of the distribution file, for the license terms of the components that this product depends on.

This product contains the extensions to Java Collections Framework which has been derived from the works by JSR-166 EG, Doug Lea, and Jason T. Greene:

* LICENSE:

* [license/LICENSE.jsr166y.txt](#) (Public Domain)

* HOMEPAGE:

* <http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/>

* <http://viewvc.jboss.org/cgi-bin/viewvc.cgi/jboss/cache/experimental/jsr166/>

This product contains a modified version of Robert Harder's Public Domain Base64 Encoder and Decoder, which can be obtained at:

* LICENSE:

* [license/LICENSE.base64.txt](#) (Public Domain)

* HOMEPAGE:

* <http://iharder.sourceforge.net/current/java/base64/>

This product contains a modified portion of 'Webbit', an event based WebSocket and HTTP server, which can be obtained at:

* LICENSE:

- * license/LICENSE.webbit.txt (BSD License)
- * HOMEPAGE:
 - * <https://github.com/joewalnes/webbit>

This product contains a modified portion of 'SLF4J', a simple logging facade for Java, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.slf4j.txt (MIT License)
- * HOMEPAGE:
 - * <https://www.slf4j.org/>

This product contains a modified portion of 'Apache Harmony', an open source Java SE, which can be obtained at:

- * NOTICE:
 - * license/NOTICE.harmony.txt
- * LICENSE:
 - * license/LICENSE.harmony.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://archive.apache.org/dist/harmony/>

This product contains a modified portion of 'jzip2', a Java bzip2 compression and decompression library written by Matthew J. Francis. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.jzip2.txt (MIT License)
- * HOMEPAGE:
 - * <https://code.google.com/p/jzip2/>

This product contains a modified portion of 'libdivsufsort', a C API library to construct the suffix array and the Burrows-Wheeler transformed string for any input string of a constant-size alphabet written by Yuta Mori. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.libdivsufsort.txt (MIT License)
- * HOMEPAGE:
 - * <https://github.com/y-256/libdivsufsort>

This product contains a modified portion of Nitsan Wakart's 'JCTools', Java Concurrency Tools for the JVM, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.jctools.txt (ASL2 License)
- * HOMEPAGE:
 - * <https://github.com/JCTools/JCTools>

This product optionally depends on 'JZlib', a re-implementation of zlib in

pure Java, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.jzlib.txt (BSD style License)
- * HOMEPAGE:
 - * <http://www.jcraft.com/jzlib/>

This product optionally depends on 'Compress-LZF', a Java library for encoding and decoding data in LZF format, written by Tatu Saloranta. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.compress-lzf.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/ning/compress>

This product optionally depends on 'lz4', a LZ4 Java compression and decompression library written by Adrien Grand. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.lz4.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/jpountz/lz4-java>

This product optionally depends on 'lzma-java', a LZMA Java compression and decompression library, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.lzma-java.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/jponge/lzma-java>

This product optionally depends on 'zstd-jni', a zstd-jni Java compression and decompression library, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.zstd-jni.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/luben/zstd-jni>

This product contains a modified portion of 'jfastlz', a Java port of FastLZ compression and decompression library written by William Kinney. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.jfastlz.txt (MIT License)
- * HOMEPAGE:
 - * <https://code.google.com/p/jfastlz/>

This product contains a modified portion of and optionally depends on 'Protocol Buffers', Google's data

interchange format, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.protobuf.txt (New BSD License)
- * HOMEPAGE:
 - * <https://github.com/google/protobuf>

This product optionally depends on 'Bouncy Castle Crypto APIs' to generate a temporary self-signed X.509 certificate when the JVM does not provide the equivalent functionality. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.bouncycastle.txt (MIT License)
- * HOMEPAGE:
 - * <https://www.bouncycastle.org/>

This product optionally depends on 'Snappy', a compression library produced by Google Inc, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.snappy.txt (New BSD License)
- * HOMEPAGE:
 - * <https://github.com/google/snappy>

This product optionally depends on 'JBoss Marshalling', an alternative Java serialization API, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.jboss-marshalling.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/jboss-remoting/jboss-marshalling>

This product optionally depends on 'Caliper', Google's micro-benchmarking framework, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.caliper.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/google/caliper>

This product optionally depends on 'Apache Commons Logging', a logging framework, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.commons-logging.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://commons.apache.org/logging/>

This product optionally depends on 'Apache Log4J', a logging framework, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.log4j.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://logging.apache.org/log4j/>

This product optionally depends on 'Aalto XML', an ultra-high performance non-blocking XML processor, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.aalto-xml.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://wiki.fasterxml.com/AaltoHome>

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Twitter. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.hpack.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/twitter/hpack>

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Cory Benfield. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.hyper-hpack.txt (MIT License)
- * HOMEPAGE:
 - * <https://github.com/python-hyper/hpack/>

This product contains a modified version of 'HPACK', a Java implementation of the HTTP/2 HPACK algorithm written by Tatsuhiro Tsujikawa. It can be obtained at:

- * LICENSE:
 - * license/LICENSE.nghttp2-hpack.txt (MIT License)
- * HOMEPAGE:
 - * <https://github.com/nghttp2/nghttp2/>

This product contains a modified portion of 'Apache Commons Lang', a Java library provides utilities for the java.lang API, which can be obtained at:

- * LICENSE:
 - * license/LICENSE.commons-lang.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://commons.apache.org/proper/commons-lang/>

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

* LICENSE:

* license/LICENSE.mvn-wrapper.txt (Apache License 2.0)

* HOMEPAGE:

* <https://github.com/takari/maven-wrapper>

This product contains the dnsinfo.h header file, that provides a way to retrieve the system DNS configuration on MacOS.

This private header is also used by Apple's open source mDNSResponder (<https://opensource.apple.com/tarballs/mDNSResponder/>).

* LICENSE:

* license/LICENSE.dnsinfo.txt (Apple Public Source License 2.0)

* HOMEPAGE:

* <https://www.opensource.apple.com/source/configd/configd-453.19/dnsinfo/dnsinfo.h>

This product optionally depends on 'Brotli4j', Brotli compression and decompression for Java., which can be obtained at:

* LICENSE:

* license/LICENSE.brotli4j.txt (Apache License 2.0)

* HOMEPAGE:

* <https://github.com/hyperxpro/Brotli4j>

Protocol Buffers - Google's data interchange format

Copyright 2013 Google Inc. All rights reserved.

<https://developers.google.com/protocol-buffers/>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Code generated by the Protocol Buffer compiler is owned by the owner of the input file used when generating it. This code is not standalone and requires a support library to be linked with it. This support library is itself covered by the above license.

/*

* Copyright (c) 2004-2007 QOS.ch

* All rights reserved.

*

* Permission is hereby granted, free of charge, to any person obtaining

* a copy of this software and associated documentation files (the

* "Software"), to deal in the Software without restriction, including

* without limitation the rights to use, copy, modify, merge, publish,

* distribute, sublicense, and/or sell copies of the Software, and to

* permit persons to whom the Software is furnished to do so, subject to

* the following conditions:

*

* The above copyright notice and this permission notice shall be

* included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE

* LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION

* OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

* WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*/

This copy of Aalto XML processor is licensed under the

Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the

specific rights regarding derivate works.

You may obtain a copy of the License at:

<https://www.apache.org/licenses/>

A copy is also included with both the the downloadable source code package

and jar that contains class bytecodes, as file "ASL 2.0". In both cases,

that file should be located next to this file: in source distribution

the location should be "release-notes/asl"; and in jar "META-INF/"

Apache Harmony

Copyright 2006, 2010 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

1.206 objenesis 3.2

1.206.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.207 micronaut-r2dbc 2.1.0

1.207.1 Available under license :

mcr.microsoft.com/mssql/server:2017-CU12

Apache License

Version 2.0, January 2004

<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.208 commons-logging 1.2

1.208.1 Available under license :

Apache Commons Logging
Copyright 2003-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.209 ktor-server-host-common 1.2.6

1.209.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.210 okhttp 3.12.12

1.210.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):

- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/OkHostnameVerifier.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/BridgeInterceptor.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/ConnectInterceptor.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/CacheInterceptor.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/OptionalMethod.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/ConnectionPool.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/RealConnection.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Connection.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Headers.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/DistinguishedNameParser.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/CertificateChainCleaner.java

No license file was found, but licenses were detected in source scan.

/*

- * Copyright (C) 2014 Square, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.

*/

Found in path(s):

- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/FormBody.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/CipherSuite.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/PushObserver.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/ws/WebSocketReader.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/RealCall.java
- * /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/Internal.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/Version.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/TlsVersion.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Call.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/ws/WebSocketProtocol.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/ResponseBody.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/ws/WebSocketWriter.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/RealResponseBody.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/MultipartBody.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Protocol.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/RequestBody.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/CertificatePinner.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/ConscryptPlatform.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/CacheRequest.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Challenge.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Callback.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Interceptor.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Header.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/HttpMethod.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/ConnectionSpec.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Credentials.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2016 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/BasicTrustRootIndex.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/CallServerInterceptor.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache2/Relay.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/UnrepeatableRequestBody.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/ConnectionShutdownException.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/Jdk9Platform.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/JdkWithJettyBootPlatform.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/WebSocketListener.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/RealInterceptorChain.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/BasicCertificateChainCleaner.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/AndroidPlatform.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/tls/TrustRootIndex.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/ws/RealWebSocket.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache2/FileOperator.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/RetryAndFollowUpInterceptor.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/StreamResetException.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/WebSocket.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/ConnectionSpecSelector.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Authenticator.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/StreamAllocation.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Cookie.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/io/FileSystem.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/HttpUrl.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/CookieJar.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/RouteException.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/FaultHidingSink.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013 Twitter, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Huffman.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 Square, Inc.

* Copyright (C) 2012 The Android Open Source Project

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/Platform.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Android Open Source Project
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2Writer.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/DiskLruCache.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2Reader.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2Stream.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/HttpDate.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2Connection.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Android Open Source Project

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Cache.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2013 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/InternalCache.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/RequestLine.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/RouteDatabase.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/ErrorCode.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Hpack.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Response.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/MediaType.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Dispatcher.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Request.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Route.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/NamedRunnable.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Handshake.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/StatusLine.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/cache/CacheStrategy.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Dns.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/connection/RouteSelector.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/OkHttpClient.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Settings.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/platform/Android10Platform.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2017 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/EventListener.java
* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/publicsuffix/PublicSuffixDatabase.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2018 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/proxy/NullProxySelector.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/annotations/EverythingIsNonNull.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 The Android Open Source Project

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http2/Http2Codec.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/Address.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/Util.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/HttpHeaders.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http/HttpCodec.java

* /opt/cola/permits/1331474321_1653068657.9097114/0/okhttp-3-12-12-sources-3-jar/okhttp3/internal/http1/Http1Codec.java

1.211 jsr311-api 1.1.1

1.211.1 Available under license :

/*

* The contents of this file are subject to the terms

* of the Common Development and Distribution License

* (the "License"). You may not use this file except

* in compliance with the License.

*

* You can obtain a copy of the license at

* <http://www.opensource.org/licenses/cddl1.php>

- * See the License for the specific language governing
- * permissions and limitations under the License.
- */

1.212 javabeans-activation-framework-api

1.2.2

1.212.1 Available under license :

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

[subs="normal"]

....

Specification: {doctitle}

Version: {revnumber}

ifeval::["{revremark}" != ""]

Status: {revremark}

endif::[]

ifeval::["{revremark}" == ""]

Status: Final Release

endif::[]

Release: {revdate}

....

Copyright (c) 2019 Eclipse Foundation.

=== Eclipse Foundation Specification License

By using and/or copying this document, or the Eclipse Foundation document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the Eclipse Foundation document from which this statement is linked, in any medium for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

- * link or URL to the original Eclipse Foundation document.
- * All existing copyright notices, or if one does not exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright (c) [date-of-document]
Eclipse Foundation, Inc. <<url to this license>>"

Inclusion of the full text of this NOTICE must be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of Eclipse Foundation documents is granted pursuant to this license, except anyone may prepare and distribute derivative works and portions of this document in software that implements the specification, in supporting materials accompanying such software, and in documentation of such software, PROVIDED that all such works include the notice below. HOWEVER, the publication of derivative works of this document for use as a technical specification is expressly prohibited.

The notice is:

"Copyright (c) 2018 Eclipse Foundation. This software or document includes material copied from or derived from [title and URI of the Eclipse Foundation specification document]."

==== Disclaimers

THIS DOCUMENT IS PROVIDED "AS IS," AND THE COPYRIGHT HOLDERS AND THE ECLIPSE FOUNDATION MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

THE COPYRIGHT HOLDERS AND THE ECLIPSE FOUNDATION WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of the copyright holders or the Eclipse Foundation may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

Notices for Jakarta Activation

This content is produced and maintained by Jakarta Activation project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jaf>

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0, which is available at <http://www.eclipse.org/org/documents/edl-v10.php>.

SPDX-License-Identifier: BSD-3-Clause

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jaf>

Third-party Content

This project leverages the following third party content.

JUnit (4.12)

* License: Eclipse Public License

1.213 arrow-annotations 0.9.0

1.213.1 Available under license :

Apache-2.0

1.214 argparse 0.8.1

1.214.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 Tatsuhiro Tsujikawa

*

* Permission is hereby granted, free of charge, to any person

* obtaining a copy of this software and associated documentation

* files (the "Software"), to deal in the Software without

* restriction, including without limitation the rights to use, copy,

* modify, merge, publish, distribute, sublicense, and/or sell copies

* of the Software, and to permit persons to whom the Software is

* furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be

* included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS

* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN

* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN

* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

* SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-

jar/net/sourceforge/argparse4j/inf/ArgumentType.java

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-

jar/net/sourceforge/argparse4j/inf/ArgumentParser.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/CJKTextWidthCounter.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/HelpArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/PrefixPattern.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/TextWidthCounter.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/UnrecognizedArgumentException.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/annotation/Arg.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/ArgumentParserException.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/StoreConstArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/ArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/Arguments.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/ArgumentGroup.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/Subparsers.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/ReflectHelper.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/UnrecognizedCommandException.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/StoreArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/Namespace.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/choice/CollectionArgumentChoice.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/ArgumentParsers.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/ArgumentParserImpl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/StoreFalseArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/AppendConstArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/ArgumentGroupImpl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/Argument.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-

jar/net/sourceforge/argparse4j/inf/ArgumentChoice.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/FeatureControl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/choice/RangeArgumentChoice.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/SubparsersImpl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/SubparserImpl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/Subparser.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/StringArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/AppendArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/ASCIITextWidthCounter.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/FileVerification.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/helper/TextHelper.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/EnumArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/StoreTrueArgumentAction.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/ConstructorArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/ParseState.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/ArgumentImpl.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/action/VersionArgumentAction.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Andrew January

*

* Permission is hereby granted, free of charge, to any person

* obtaining a copy of this software and associated documentation

* files (the "Software"), to deal in the Software without

* restriction, including without limitation the rights to use, copy,

* modify, merge, publish, distribute, sublicense, and/or sell copies

* of the Software, and to permit persons to whom the Software is

* furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be

* included in all copies or substantial portions of the Software.

*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/CaseInsensitiveEnumStringArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/CaseInsensitiveEnumNameArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/EnumStringArgumentType.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Tatsuhiro Tsujikawa

*

* Permission is hereby granted, free of charge, to any person
* obtaining a copy of this software and associated documentation
* files (the "Software"), to deal in the Software without
* restriction, including without limitation the rights to use, copy,
* modify, merge, publish, distribute, sublicense, and/or sell copies
* of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/BooleanArgumentType.java
* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-

jar/net/sourceforge/argparse4j/inf/MetavarInference.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one
* or more contributor license agreements. See the NOTICE file
* distributed with this work for additional information
* regarding copyright ownership. The ASF licenses this file
* to you under the Apache License, Version 2.0 (the
* "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing,
* software distributed under the License is distributed on an
* "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY
* KIND, either express or implied. See the License for the
* specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/impl/type/FileArgumentType.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 Adam Parkin
*
* Permission is hereby granted, free of charge, to any person
* obtaining a copy of this software and associated documentation
* files (the "Software"), to deal in the Software without
* restriction, including without limitation the rights to use, copy,
* modify, merge, publish, distribute, sublicense, and/or sell copies
* of the Software, and to permit persons to whom the Software is
* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

* SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-jar/net/sourceforge/argparse4j/helper/HelpScreenException.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 Tatsuhiro Tsujikawa

*

* Permission is hereby granted, free of charge, to any person

* obtaining a copy of this software and associated documentation

* files (the "Software"), to deal in the Software without

* restriction, including without limitation the rights to use, copy,

* modify, merge, publish, distribute, sublicense, and/or sell copies

* of the Software, and to permit persons to whom the Software is

* furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be

* included in all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,

* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF

* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND

* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS

* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN

* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN

* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE

* SOFTWARE.

*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-jar/net/sourceforge/argparse4j/impl/action/CountArgumentAction.java

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-jar/net/sourceforge/argparse4j/impl/type/ReflectArgumentType.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 Tatsuhiro Tsujikawa

*

* Permission is hereby granted, free of charge, to any person

* obtaining a copy of this software and associated documentation

* files (the "Software"), to deal in the Software without

* restriction, including without limitation the rights to use, copy,

* modify, merge, publish, distribute, sublicense, and/or sell copies

* of the Software, and to permit persons to whom the Software is

* furnished to do so, subject to the following conditions:
*
* The above copyright notice and this permission notice shall be
* included in all copies or substantial portions of the Software.
*
* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
* EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
* MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND
* NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS
* BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN
* ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN
* CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
* SOFTWARE.
*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/inf/MutuallyExclusiveGroup.java
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274701436_1645235201.06/0/argparse4j-0-8-1-sources-
jar/net/sourceforge/argparse4j/internal/TerminalWidth.java

1.215 opentelemetry-java 1.6.0

1.215.1 Available under license :

c-libutl 20160225

c-libutl License

...

This software is distributed under the terms of the BSD license.

== BSD LICENSE =====

(C) 2009 by Remo Dentato (rdentato@gmail.com)

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<http://opensource.org/licenses/bsd-license.php>

...

IAIK (Institute for Applied Information Processing and Communication) PKCS#11 wrapper files v1

IAIK License

<pre>

Copyright (c) 2002 Graz University of Technology. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by IAIK of Graz University of Technology."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names "Graz University of Technology" and "IAIK of Graz University of Technology" must not be used to endorse or promote products derived from this software without prior written permission.

5. Products derived from this software may not be called "IAIK PKCS Wrapper", nor may "IAIK" appear in their name, without prior written permission of Graz University of Technology.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE LICENSOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

This copy of Jackson JSON processor YAML module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivative works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

This library is free software; you can redistribute it and/or modify it under the terms of the Modified BSD License.

The complete text of the license is available at the Documentation/licenses/COPYING.BSD-3 file.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.

b) Accompany the Combined Work with a copy of the GNU GPL and this license document.

c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.

d) Do one of the following:

0) Convey the Minimal Corresponding Source under the terms of this

License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

SPECIAL EXCEPTION GRANTED BY COPYRIGHT HOLDERS

As a special exception, copyright holders give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you must extend this exception to your version of the library.

Double-conversion v1.1.5

Double-conversion License

<https://raw.githubusercontent.com/google/double-conversion/master/LICENSE>

<pre>

Copyright 2006-2011, the V8 project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
</pre>
### Apache Xerces v2.12.1
```

```
### Apache Xerces Notice
```

```
<pre>
=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====
```

Apache Xerces Java
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

```
</pre>

### Apache 2.0 License
<pre>
```

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

</pre>

Eastman Kodak Company: Portions of color management and imaging software

Eastman Kodak Notice

<pre>

Portions Copyright Eastman Kodak Company 1991-2003

</pre>

End User License Agreement Missing

Apache HttpClient Cache

Copyright 2010-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====

Apache Xerces Java
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the
Apache Software Foundation that were originally developed at iClick, Inc.,
software copyright (c) 1999.

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates
the terms and conditions of version 3 of the GNU General Public
License, supplemented by the additional permissions listed below.

0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

A "Combined Work" is a work produced by combining or linking an Application with the Library. The particular version of the Library with which the Combined Work was made is also called the "Linked Version".

The "Minimal Corresponding Source" for a Combined Work means the Corresponding Source for the Combined Work, excluding any source code for portions of the Combined Work that, considered in isolation, are based on the Application, and not on the Linked Version.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work.

1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL.

2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or

b) under the GNU GPL, with none of the additional permissions of

this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

- a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

- a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License.
- b) Accompany the Combined Work with a copy of the GNU GPL and this license document.
- c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document.
- d) Do one of the following:
 - 0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.

1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version.

e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License.

b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you

received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.

SPECIAL EXCEPTION GRANTED BY IMATIX

As a special exception, iMatix gives you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you must extend this exception to your version of the library.

Parts of the software are licensed under the MIT (X11) license as follows:

Copyright (c) 2007-2010 Contributors as listed in AUTHORS

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR

OTHER DEALINGS IN THE SOFTWARE.

Apache Xalan v2.7.2

Apache Xalan Notice

<pre>

```
=====
====
== NOTICE file corresponding to the section 4d of the Apache License, Version 2.0, ==
== in this case for the Apache Xalan distribution.                               ==
=====
====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

Specifically, we only include the XSLTC portion of the source from the Xalan distribution.
The Xalan project has two processors: an interpretive one (Xalan Interpretive) and a
compiled one (The XSLT Compiler (XSLTC)). We *only* use the XSLTC part of Xalan; We use
the source from the packages that are part of the XSLTC sources.

Portions of this software was originally based on the following:

- software copyright (c) 1999-2002, Lotus Development Corporation., <http://www.lotus.com>.
- software copyright (c) 2001-2002, Sun Microsystems., <http://www.sun.com>.
- software copyright (c) 2003, IBM Corporation., <http://www.ibm.com>.
- voluntary contributions made by Ovidiu Predescu (ovidiu@cup.hp.com) on behalf of the
Apache Software Foundation and was originally developed at Hewlett Packard Company.

</pre>

Apache 2.0 License

<pre>

```
        Apache License
        Version 2.0, January 2004
        http://www.apache.org/licenses/
```

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

JLEX COPYRIGHT NOTICE, LICENSE AND DISCLAIMER.

Copyright 1996-2003 by Elliot Joel Berk and C. Scott Ananian
Permission to use, copy, modify, and distribute this software and

its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the name of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software. The portions of JLex output which are hard-coded into the JLex source code are (naturally) covered by this same license.

</pre>

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts.

OpenSSL License

```
/* =====
```

```
* Copyright (c) 1998-2019 The OpenSSL Project. All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without  
* modification, are permitted provided that the following conditions  
* are met:
```

```
*
```

```
* 1. Redistributions of source code must retain the above copyright  
* notice, this list of conditions and the following disclaimer.
```

```
*
```

```
* 2. Redistributions in binary form must reproduce the above copyright  
* notice, this list of conditions and the following disclaimer in  
* the documentation and/or other materials provided with the  
* distribution.
```

```
*
```

```
* 3. All advertising materials mentioning features or use of this  
* software must display the following acknowledgment:  
* "This product includes software developed by the OpenSSL Project  
* for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
```

*
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS'' AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *
 * This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,

* lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 * must display the following acknowledgement:
 * "This product includes cryptographic software written by
 * Eric Young (eay@cryptsoft.com)"
 * The word 'cryptographic' can be left out if the routines from the library
 * being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 * the apps directory (application code) you must include an acknowledgement:
 * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * The licence and distribution terms for any publically available version or
 * derivative of this code cannot be changed. i.e. this code cannot simply be
 * copied and put under another distribution licence
 * [including the GNU Public Licence.]
 */
 xml-commons/java/external/README.dom.txt \$Id: README.dom.txt 477038 2006-11-20 04:40:36Z mrglavas \$

HEAR YE, HEAR YE!

All of the .java software and associated documentation about the DOM in this repository are distributed under the license from the W3C, which is provided herein.

LICENSE.dom-software.txt covers all software from the W3C including the following items in the xml-commons project:

xml-commons/java/external/src/org/w3c
and all subdirectories

(Note: SAC (The Simple API for CSS) has been published under an older version of the W3C license. The original license file is LICENSE.sac.html.)

LICENSE.dom-documentation.txt covers all documentation from the W3C including the following items in the xml-commons project:

xml-commons/java/external/xdocs/dom
and all subdirectories

The actual DOM Java Language Binding classes in xml-commons came from:

<http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.html>

The specification of DOM Level 3's various parts is at:

<http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/>
<http://www.w3.org/TR/2004/REC-DOM-Level-3-LS-20040407/>
<http://www.w3.org/TR/2004/NOTE-DOM-Level-3-XPath-20040226/>

The specification of DOM Level 2's various parts is at:

<http://www.w3.org/TR/2000/REC-DOM-Level-2-Events-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Style-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Traversal-Range-20001113/>
<http://www.w3.org/TR/2000/REC-DOM-Level-2-Views-20001113/>

The specification of DOM Level 1's various parts is at:

<http://www.w3.org/TR/1998/REC-DOM-Level-1-19981001/level-one-html.html>

Links to all available W3C DOM Java Bindings can be found at:

<http://www.w3.org/DOM/DOMTR>

The actual classes of The Simple API for CSS (SAC) came from:

<http://www.w3.org/Style/CSS/SAC/>
<http://www.w3.org/2002/06/sacjava-1.3.zip>

The actual DOM Java Language Binding classes for SMIL came from:

<http://dev.w3.org/cvsweb/java/classes/org/w3c/dom/smil/>

(both `ElementTimeControl.java` and `TimeEvent.java` were taken at revision 1.1)

The actual DOM Java Language Binding classes for SVG 1.1 came from:

<http://www.w3.org/TR/SVG11/java.html>

Apache Commons IO

Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

OPENJDK ASSEMBLY EXCEPTION

The OpenJDK source code made available by Oracle America, Inc. (Oracle) at openjdk.java.net ("OpenJDK Code") is distributed under the terms of the GNU General Public License <<http://www.gnu.org/copyleft/gpl.html>> version 2 only ("GPL2"), with the following clarification and special exception.

Linking this OpenJDK Code statically or dynamically with other code is making a combined work based on this library. Thus, the terms and conditions of GPL2 cover the whole combination.

As a special exception, Oracle gives you permission to link this OpenJDK Code with certain code licensed by Oracle as indicated at <http://openjdk.java.net/legal/exception-modules-2007-05-08.html> ("Designated Exception Modules") to produce an executable, regardless of the license terms of the Designated Exception Modules, and to copy and distribute the resulting executable under GPL2, provided that the Designated Exception Modules continue to be governed by the licenses under which they were offered by Oracle.

As such, it allows licensees and sublicensees of Oracle's GPL2 OpenJDK Code to build an executable that includes those portions of necessary code that Oracle could not provide under GPL2 (or that Oracle has provided under GPL2 with the Classpath exception). If you modify or add to the OpenJDK code, that new GPL2 code may still be combined with Designated Exception Modules if the new code is made subject to this exception by its copyright holder.

PC/SC Lite v1.8.26

PC/SC Lite License

<pre>

Copyright (c) 1999-2003 David Corcoran <corcoran@linuxnet.com>

Copyright (c) 2001-2011 Ludovic Rousseau <ludovic.rousseau@free.fr>

All rights reserved.

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Changes to this license can be made only by the copyright author with explicit written consent.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

Independent JPEG Group: JPEG release 6b

JPEG License

<pre>

Must reproduce following license in documentation and/or other materials provided with distribution:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you, its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

- (1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty

notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.

(2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".

(3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

ansi2knr.c is included in this distribution by permission of L. Peter Deutsch, sole proprietor of its copyright holder, Aladdin Enterprises of Menlo Park, CA. ansi2knr.c is NOT covered by the above copyright and conditions, but instead by the usual distribution terms of the Free Software Foundation; principally, that you must include source code if you redistribute it.

(See the file ansi2knr.c for full details.) However, since ansi2knr.c is not needed as part of any program generated from the IJG code, this does not limit you more than the foregoing paragraphs do.

The Unix configuration script "configure" was produced with GNU Autoconf. It is copyright by the Free Software Foundation but is freely distributable. The same holds for its supporting scripts (config.guess, config.sub, ltconfig, ltmain.sh). Another support script, install-sh, is copyright by M.I.T. but is also freely distributable.

It appears that the arithmetic coding option of the JPEG spec is covered by patents owned by IBM, AT&T, and Mitsubishi. Hence arithmetic coding cannot legally be used without obtaining one or more licenses. For this reason, support for arithmetic coding has been removed from the free JPEG software. (Since arithmetic coding provides only a marginal gain over the unpatented Huffman mode, it is unlikely that very many implementations will support it.) So far as we are aware, there are no patent restrictions on the remaining code.

The IIG distribution formerly included code to read and write GIF files. To avoid entanglement with the Unisys LZW patent, GIF reading support has been removed altogether, and the GIF writer has been simplified to produce "uncompressed GIFs". This technique does not use the LZW algorithm; the resulting GIF files are larger than usual, but are readable by all standard GIF decoders.

We are required to state that "The Graphics Interchange Format(c) is the Copyright property of CompuServe Incorporated. GIF(sm) is a Service Mark property of CompuServe Incorporated."

</pre>

```
=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache xml-commons xml-apis ==
== distribution. ==
=====
```

Apache XML Commons XML APIs
Copyright 1999-2009 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:
- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- software copyright (c) 2000 World Wide Web Consortium, <http://www.w3.org>
DOM Level 3 Core Specification v1.0

W3C License

<pre>

W3C SOFTWARE NOTICE AND LICENSE

<http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including

modifications:

- 1.The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
- 2.Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
- 3.Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS,COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION. The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". Otherwise, this version is the same as the previous version and is written so as to preserve the Free Software Foundation's assessment of GPL compatibility and OSI's certification under the Open Source Definition. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

</pre>

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

- 1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You

distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of

the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING

NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and

every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions.

You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE

STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

The Netty Project
=====

Please visit the Netty web site for more information:

* <http://netty.io/>

Copyright 2016 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a forked and modified version of Tomcat Native

- * LICENSE:
 - * license/LICENSE.tomcat-native.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <http://tomcat.apache.org/native-doc/>
 - * <https://svn.apache.org/repos/asf/tomcat/native/>

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

- * LICENSE:
 - * license/LICENSE.mvn-wrapper.txt (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/takari/maven-wrapper>

This product contains small piece of code to support AIX, taken from netbsd.

- * LICENSE:
 - * license/LICENSE.aix-netbsd.txt (OpenSSL License)
- * HOMEPAGE:
 - * <https://ftp.netbsd.org/pub/NetBSD/NetBSD-current/src/crypto/external/bsd/openssl/dist>

This product contains code from boringssl.

- * LICENSE (Combination ISC and OpenSSL license)
 - * license/LICENSE.boringssl.txt (Combination ISC and OpenSSL license)
- * HOMEPAGE:
 - * <https://boringssl.googlesource.com/boringssl/>

The GNU General Public License (GPL)

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for

everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the

most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making

modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR

THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author Gnomovision comes
with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free
software, and you are welcome to redistribute it under certain conditions;
type 'show c' for details.
```

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than 'show w' and 'show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
signature of Ty Coon, 1 April 1989
```

```
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL

Certain source files distributed by Oracle America and/or its affiliates are subject to the following clarification and special exception to the GPL, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an

executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

```
## Dynalink v.5
```

```
### Dynalink License
```

```
<pre>
```

Copyright (c) 2009-2013, Attila Szegedi

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the copyright holder nor the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL COPYRIGHT HOLDER BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
</pre>
```

```
xml-commons/java/external/LICENSE.dom-software.txt $Id: LICENSE.dom-software.txt 734314 2009-01-14 03:33:27Z mrglavas $
```

This license came from: <http://www.w3.org/TR/2004/REC-DOM-Level-3-Core-20040407/java-binding.zip> (COPYRIGHT.html)

W3C SOFTWARE NOTICE AND LICENSE

Copyright 2004 World Wide Web Consortium, (Massachusetts Institute of Technology,

European Research Consortium for Informatics and Mathematics, Keio University).
All Rights Reserved.

The DOM bindings are published under the W3C Software Copyright Notice and License. The software license requires "Notice of any changes or modifications to the W3C files, including the date changes were made." Consequently, modified versions of the DOM bindings must document that they do not conform to the W3C standard; in the case of the IDL definitions, the pragma prefix can no longer be 'w3c.org'; in the case of the Java language binding, the package names can no longer be in the 'org.w3c' package.

Note: The original version of the W3C Software Copyright Notice and License could be found at <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>

This work (and included software, documentation such as READMEs, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions.

Permission to copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications:

1. The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.
2. Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, the W3C Software Short Notice should be included (hypertext is preferred, text is permitted) within the body of any redistributed or derivative code.
3. Notice of any changes or modifications to the files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission.

Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

Apache Commons Math
Copyright 2001-2010 The Apache Software Foundation

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

=====
The LinearConstraint, LinearObjectiveFunction, LinearOptimizer,
Relationship, SimplexSolver and SimplexTableau classes in package
org.apache.commons.math.optimization.linear include software developed by
Benjamin McCann (<http://www.benmccann.com>) and distributed with
the following copyright: Copyright 2009 Google Inc.
=====

This product includes software developed by the
University of Chicago, as Operator of Argonne National
Laboratory.
The LevenbergMarquardtOptimizer class in package
org.apache.commons.math.optimization.general includes software
translated from the lmdcr, lmpar and qrsolv Fortran routines
from the Minpack package
Minpack Copyright Notice (1999) University of Chicago. All rights reserved
=====

The GraggBulirschStoerIntegrator class in package
org.apache.commons.math.ode.nonstiff includes software translated
from the odex Fortran routine developed by E. Hairer and G. Wanner.
Original source copyright:
Copyright (c) 2004, Ernst Hairer
=====

The EigenDecompositionImpl class in package
org.apache.commons.math.linear includes software translated
from some LAPACK Fortran routines. Original source copyright:
Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.
=====

The MersenneTwister class in package org.apache.commons.math.random
includes software translated from the 2002-01-26 version of
the Mersenne-Twister generator written in C by Makoto Matsumoto and Takuji
Nishimura. Original source copyright:
Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura,
All rights reserved
=====

The complete text of licenses and disclaimers associated with the the original
sources enumerated above at the time of code translation are in the LICENSE.txt
file.

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium:
<https://cla.developers.google.com/clas>

Files in `third_party/` have their own licenses, as described therein. The MIT license, for `third_party/fiat`, which, unlike other `third_party` directories, is compiled into non-test libraries, is included below.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

The following are Google-internal bug numbers where explicit permission from some authors is recorded for use of their work. (This is purely for our own record keeping.)

27287199

27287880

27287883

OpenSSL License

```
/* =====
```

```
* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without  
* modification, are permitted provided that the following conditions  
* are met:
```

```
*
```

```
* 1. Redistributions of source code must retain the above copyright  
* notice, this list of conditions and the following disclaimer.
```

```
*
```

```
* 2. Redistributions in binary form must reproduce the above copyright  
* notice, this list of conditions and the following disclaimer in  
* the documentation and/or other materials provided with the  
* distribution.
```

```
*
```

```
* 3. All advertising materials mentioning features or use of this  
* software must display the following acknowledgment:  
* "This product includes software developed by the OpenSSL Project  
* for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
```

```
*
```

```
* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to  
* endorse or promote products derived from this software without  
* prior written permission. For written permission, please contact  
* openssl-core@openssl.org.
```

```
*
```

```
* 5. Products derived from this software may not be called "OpenSSL"  
* nor may "OpenSSL" appear in their names without prior written
```

```

* permission of the OpenSSL Project.
*
* 6. Redistributions of any form whatsoever must retain the following
* acknowledgment:
* "This product includes software developed by the OpenSSL Project
* for use in the OpenSSL Toolkit (http://www.openssl.org/)"
*
* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
* EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
* LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
* STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
* OF THE POSSIBILITY OF SUCH DAMAGE.
* =====
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com). This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/
Original SSLeay License
-----
/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
* All rights reserved.
*
* This package is an SSL implementation written
* by Eric Young (eay@cryptsoft.com).
* The implementation was written so as to conform with Netscapes SSL.
*
* This library is free for commercial and non-commercial use as long as
* the following conditions are aheared to. The following conditions
* apply to all code found in this distribution, be it the RC4, RSA,
* lhash, DES, etc., code; not just the SSL code. The SSL documentation
* included with this distribution is covered by the same copyright terms
* except that the holder is Tim Hudson (tjh@cryptsoft.com).
*
* Copyright remains Eric Young's, and as such any Copyright notices in
* the code are not to be removed.
* If this package is used in a product, Eric Young should be given attribution
* as the author of the parts of the library used.
* This can be in the form of a textual message at program startup or
* in documentation (online or textual) provided with the package.

```

*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. All advertising materials mentioning features or use of this software
* must display the following acknowledgement:
* "This product includes cryptographic software written by
* Eric Young (eay@cryptsoft.com)"
* The word 'cryptographic' can be left out if the routines from the library
* being used are not cryptographic related :-).
* 4. If you include any Windows specific code (or a derivative thereof) from
* the apps directory (application code) you must include an acknowledgement:
* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
*
* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
*
* The licence and distribution terms for any publically available version or
* derivative of this code cannot be changed. i.e. this code cannot simply be
* copied and put under another distribution licence
* [including the GNU Public Licence.]
*/
ISC license used for completely new code in BoringSSL:
/* Copyright (c) 2015, Google Inc.
*
* Permission to use, copy, modify, and/or distribute this software for any
* purpose with or without fee is hereby granted, provided that the above
* copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
* SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION

* OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
* CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

The code in third_party/flat carries the MIT license:

Copyright (c) 2015-2016 the fiat-crypto authors (see
<https://github.com/mit-plv/flat-crypto/blob/master/AUTHORS>).

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Licenses for support code

Parts of the TLS test suite are under the Go license. This code is not included
in BoringSSL (i.e. libcrypto and libssl) when compiled, however, so
distributing code linked against BoringSSL does not trigger this license:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

- * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE

OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BoringSSL uses the Chromium test infrastructure to run a continuous build, trybots etc. The scripts which manage this, and the script for generating build metadata, are under the Chromium license. Distributing code linked against BoringSSL does not trigger this license.

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Mozilla Public Suffix List

Public Suffix Notice

^^^

You are receiving a copy of the Mozilla Public Suffix List in the following file: <java-home>/lib/security/public_suffix_list.dat. The terms of the Oracle license do NOT apply to this file; it is licensed under the Mozilla Public License 2.0, separately from the Oracle programs you receive. If you do not wish to use the Public Suffix List, you may remove the <java-home>/lib/security/public_suffix_list.dat file.

The Source Code of this file is available under the Mozilla Public License, v. 2.0 and is located at

https://raw.githubusercontent.com/publicsuffix/list/cbbba1d234670453df9c930dfbf510c0474d4301/public_suffix_list.dat.

If a copy of the MPL was not distributed with this file, you can obtain one at <https://mozilla.org/MPL/2.0/>.

Software distributed under the License is distributed on an "AS IS" basis,

WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

...

MPL v2.0

...

Mozilla Public License Version 2.0

=====

1. Definitions

1.1. "Contributor"

means each individual or legal entity that creates, contributes to the creation of, or owns Covered Software.

1.2. "Contributor Version"

means the combination of the Contributions of others (if any) used by a Contributor and that particular Contributor's Contribution.

1.3. "Contribution"

means Covered Software of a particular Contributor.

1.4. "Covered Software"

means Source Code Form to which the initial Contributor has attached the notice in Exhibit A, the Executable Form of such Source Code Form, and Modifications of such Source Code Form, in each case including portions thereof.

1.5. "Incompatible With Secondary Licenses"

means

(a) that the initial Contributor has attached the notice described in Exhibit B to the Covered Software; or

(b) that the Covered Software was made available under the terms of version 1.1 or earlier of the License, but not also under the terms of a Secondary License.

1.6. "Executable Form"

means any form of the work other than Source Code Form.

1.7. "Larger Work"

means a work that combines Covered Software with other material, in a separate file or files, that is not Covered Software.

1.8. "License"

means this document.

1.9. "Licensable"

means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently, any and all of the rights conveyed by this License.

1.10. "Modifications"

means any of the following:

(a) any file in Source Code Form that results from an addition to, deletion from, or modification of the contents of Covered Software; or

(b) any new file in Source Code Form that contains any Covered Software.

1.11. "Patent Claims" of a Contributor

means any patent claim(s), including without limitation, method, process, and apparatus claims, in any patent Licensable by such Contributor that would be infringed, but for the grant of the License, by the making, using, selling, offering for sale, having made, import, or transfer of either its Contributions or its Contributor Version.

1.12. "Secondary License"

means either the GNU General Public License, Version 2.0, the GNU Lesser General Public License, Version 2.1, the GNU Affero General Public License, Version 3.0, or any later versions of those licenses.

1.13. "Source Code Form"

means the form of the work preferred for making modifications.

1.14. "You" (or "Your")

means an individual or a legal entity exercising rights under this License. For legal entities, "You" includes any entity that controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants and Conditions

2.1. Grants

Each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by such Contributor to use, reproduce, make available, modify, display, perform, distribute, and otherwise exploit its Contributions, either on an unmodified basis, with Modifications, or as part of a Larger Work; and
- (b) under Patent Claims of such Contributor to make, use, sell, offer for sale, have made, import, and otherwise transfer either its Contributions or its Contributor Version.

2.2. Effective Date

The licenses granted in Section 2.1 with respect to any Contribution become effective for each Contribution on the date the Contributor first distributes such Contribution.

2.3. Limitations on Grant Scope

The licenses granted in this Section 2 are the only rights granted under this License. No additional rights or licenses will be implied from the distribution or licensing of Covered Software under this License. Notwithstanding Section 2.1(b) above, no patent license is granted by a Contributor:

- (a) for any code that a Contributor has removed from Covered Software; or
- (b) for infringements caused by: (i) Your and any other third party's modifications of Covered Software, or (ii) the combination of its Contributions with other software (except as part of its Contributor Version); or
- (c) under Patent Claims infringed by Covered Software in the absence of its Contributions.

This License does not grant any rights in the trademarks, service marks, or logos of any Contributor (except as may be necessary to comply with the notice requirements in Section 3.4).

2.4. Subsequent Licenses

No Contributor makes additional grants as a result of Your choice to distribute the Covered Software under a subsequent version of this License (see Section 10.2) or under the terms of a Secondary License (if permitted under the terms of Section 3.3).

2.5. Representation

Each Contributor represents that the Contributor believes its Contributions are its original creation(s) or it has sufficient rights to grant the rights to its Contributions conveyed by this License.

2.6. Fair Use

This License is not intended to limit any rights You have under applicable copyright doctrines of fair use, fair dealing, or other equivalents.

2.7. Conditions

Sections 3.1, 3.2, 3.3, and 3.4 are conditions of the licenses granted in Section 2.1.

3. Responsibilities

3.1. Distribution of Source Form

All distribution of Covered Software in Source Code Form, including any Modifications that You create or to which You contribute, must be under the terms of this License. You must inform recipients that the Source Code Form of the Covered Software is governed by the terms of this License, and how they can obtain a copy of this License. You may not attempt to alter or restrict the recipients' rights in the Source Code Form.

3.2. Distribution of Executable Form

If You distribute Covered Software in Executable Form then:

- (a) such Covered Software must also be made available in Source Code Form, as described in Section 3.1, and You must inform recipients of the Executable Form how they can obtain a copy of such Source Code Form by reasonable means in a timely manner, at a charge no more than the cost of distribution to the recipient; and
- (b) You may distribute such Executable Form under the terms of this License, or sublicense it under different terms, provided that the license for the Executable Form does not attempt to limit or alter the recipients' rights in the Source Code Form under this License.

3.3. Distribution of a Larger Work

You may create and distribute a Larger Work under terms of Your choice, provided that You also comply with the requirements of this License for the Covered Software. If the Larger Work is a combination of Covered Software with a work governed by one or more Secondary Licenses, and the Covered Software is not Incompatible With Secondary Licenses, this License permits You to additionally distribute such Covered Software under the terms of such Secondary License(s), so that the recipient of the Larger Work may, at their option, further distribute the Covered Software under the terms of either this License or such Secondary License(s).

3.4. Notices

You may not remove or alter the substance of any license notices (including copyright notices, patent notices, disclaimers of warranty, or limitations of liability) contained within the Source Code Form of the Covered Software, except that You may alter any license notices to the extent required to remedy known factual inaccuracies.

3.5. Application of Additional Terms

You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, You may do so only on Your own behalf, and not on behalf of any Contributor. You must make it absolutely clear that any such warranty, support, indemnity, or liability obligation is offered by You alone, and You hereby agree to indemnify every Contributor for any liability incurred by such Contributor as a result of warranty, support, indemnity or liability terms You offer. You may include additional disclaimers of warranty and limitations of liability specific to any jurisdiction.

4. Inability to Comply Due to Statute or Regulation

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Software due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be placed in a text file included with all distributions of the Covered Software under this License. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Termination

5.1. The rights granted under this License will terminate automatically if You fail to comply with any of its terms. However, if You become compliant, then the rights granted under this License from a particular Contributor are reinstated (a) provisionally, unless and until such Contributor explicitly and finally terminates Your grants, and (b) on an ongoing basis, if such Contributor fails to notify You of the non-compliance by some reasonable means prior to 60 days after You have come back into compliance. Moreover, Your grants from a particular Contributor are reinstated on an ongoing basis if such Contributor notifies You of the non-compliance by some reasonable means, this is the first time You have received notice of non-compliance with this License from such Contributor, and You become compliant prior to 30 days after Your receipt of the notice.

5.2. If You initiate litigation against any entity by asserting a patent infringement claim (excluding declaratory judgment actions, counter-claims, and cross-claims) alleging that a Contributor Version directly or indirectly infringes any patent, then the rights granted to You by any and all Contributors for the Covered Software under Section 2.1 of this License shall terminate.

5.3. In the event of termination under Sections 5.1 or 5.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or Your distributors under this License prior to termination shall survive termination.

```
*
*
* 6. Disclaimer of Warranty
* -----
*
* Covered Software is provided under this License on an "as is"
* basis, without warranty of any kind, either expressed, implied, or
* statutory, including, without limitation, warranties that the
* Covered Software is free of defects, merchantable, fit for a
* particular purpose or non-infringing. The entire risk as to the
* quality and performance of the Covered Software is with You.
* Should any Covered Software prove defective in any respect, You
* (not any Contributor) assume the cost of any necessary servicing,
* repair, or correction. This disclaimer of warranty constitutes an
* essential part of this License. No use of any Covered Software is
* authorized under this License except under this disclaimer.
*
```

```
*
*
* 7. Limitation of Liability
*
```


* ----- *

* *

* Under no circumstances and under no legal theory, whether tort *

* (including negligence), contract, or otherwise, shall any *

* Contributor, or anyone who distributes Covered Software as *

* permitted above, be liable to You for any direct, indirect, *

* special, incidental, or consequential damages of any character *

* including, without limitation, damages for lost profits, loss of *

* goodwill, work stoppage, computer failure or malfunction, or any *

* and all other commercial damages or losses, even if such party *

* shall have been informed of the possibility of such damages. This *

* limitation of liability shall not apply to liability for death or *

* personal injury resulting from such party's negligence to the *

* extent applicable law prohibits such limitation. Some *

* jurisdictions do not allow the exclusion or limitation of *

* incidental or consequential damages, so this exclusion and *

* limitation may not apply to You. *

* *

8. Litigation

Any litigation relating to this License may be brought only in the courts of a jurisdiction where the defendant maintains its principal place of business and such litigation shall be governed by laws of that jurisdiction, without reference to its conflict-of-law provisions. Nothing in this Section shall prevent a party's ability to bring cross-claims or counter-claims.

9. Miscellaneous

This License represents the complete agreement concerning the subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not be used to construe this License against a Contributor.

10. Versions of the License

10.1. New Versions

Mozilla Foundation is the license steward. Except as provided in Section 10.3, no one other than the license steward has the right to modify or publish new versions of this License. Each version will be given a

distinguishing version number.

10.2. Effect of New Versions

You may distribute the Covered Software under the terms of the version of the License under which You originally received the Covered Software, or under the terms of any subsequent version published by the license steward.

10.3. Modified Versions

If you create software not governed by this License, and you want to create a new license for such software, you may create and use a modified version of this License if you rename the license and remove any references to the name of the license steward (except to note that such modified license differs from this License).

10.4. Distributing Source Code Form that is Incompatible With Secondary Licenses

If You choose to distribute Source Code Form that is Incompatible With Secondary Licenses under the terms of this version of the License, the notice described in Exhibit B of this License must be attached.

Exhibit A - Source Code Form License Notice

This Source Code Form is subject to the terms of the Mozilla Public License, v. 2.0. If a copy of the MPL was not distributed with this file, You can obtain one at <https://mozilla.org/MPL/2.0/>.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Exhibit B - "Incompatible With Secondary Licenses" Notice

This Source Code Form is "Incompatible With Secondary Licenses", as defined by the Mozilla Public License, v. 2.0.

The FreeType Project: Freetype v2.10.4

FreeType Notice

...

FreeType comes with two licenses from which you can choose the one which fits your needs best.

The FreeType License (FTL) is the most commonly used one. It is a BSD-style license with a credit clause and thus compatible with the GNU Public License (GPL) version 3, but not with the GPL version 2.

The GNU General Public License (GPL), version 2. Use it for all projects which use the GPLv2 also, or which need a license compatible to the GPLv2.

...

FreeType License

...

The FreeType Project LICENSE

2006-Jan-27

Copyright 1996-2002, 2006 by
David Turner, Robert Wilhelm, and Werner Lemberg

Introduction

=====

The FreeType Project is distributed in several archive packages; some of them may contain, in addition to the FreeType font engine, various tools and contributions which rely on, or relate to, the FreeType Project.

This license applies to all files found in such packages, and which do not fall under their own explicit license. The license affects thus the FreeType font engine, the test programs, documentation and makefiles, at the very least.

This license was inspired by the BSD, Artistic, and IJG (Independent JPEG Group) licenses, which all encourage inclusion and use of free software in commercial and freeware products alike. As a consequence, its main points are that:

- o We don't promise that this software works. However, we will be

interested in any kind of bug reports. (as is' distribution)

- o You can use this software for whatever you want, in parts or full form, without having to pay us. (royalty-free' usage)
- o You may not pretend that you wrote this software. If you use it, or only parts of it, in a program, you must acknowledge somewhere in your documentation that you have used the FreeType code. (credits')

We specifically permit and encourage the inclusion of this software, with or without modifications, in commercial products. We disclaim all warranties covering The FreeType Project and assume no liability related to The FreeType Project.

Finally, many people asked us for a preferred form for a credit/disclaimer to use in compliance with this license. We thus encourage you to use the following text:

```
""  
Portions of this software are copyright <year> The FreeType  
Project (www.freetype.org). All rights reserved.  
""
```

Please replace <year> with the value from the FreeType version you actually use.

Legal Terms

=====

0. Definitions

Throughout this license, the terms 'package', 'FreeType Project', and 'FreeType archive' refer to the set of files originally distributed by the authors (David Turner, Robert Wilhelm, and Werner Lemberg) as the 'FreeType Project', be they named as alpha, beta or final release.

'You' refers to the licensee, or person using the project, where 'using' is a generic term including compiling the project's source code as well as linking it to form a 'program' or 'executable'. This program is referred to as 'a program using the FreeType engine'.

This license applies to all files distributed in the original

FreeType Project, including all source code, binaries and documentation, unless otherwise stated in the file in its original, unmodified form as distributed in the original archive. If you are unsure whether or not a particular file is covered by this license, you must contact us to verify this.

The FreeType Project is copyright (C) 1996-2000 by David Turner, Robert Wilhelm, and Werner Lemberg. All rights reserved except as specified below.

1. No Warranty

THE FREETYPE PROJECT IS PROVIDED 'AS IS' WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL ANY OF THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY DAMAGES CAUSED BY THE USE OR THE INABILITY TO USE, OF THE FREETYPE PROJECT.

2. Redistribution

This license grants a worldwide, royalty-free, perpetual and irrevocable right and license to use, execute, perform, compile, display, copy, create derivative works of, distribute and sublicense the FreeType Project (in both source and object code forms) and derivative works thereof for any purpose; and to authorize others to exercise some or all of the rights granted herein, subject to the following conditions:

- o Redistribution of source code must retain this license file ('FTL.TXT') unaltered; any additions, deletions or changes to the original files must be clearly indicated in accompanying documentation. The copyright notices of the unaltered, original files must be preserved in all copies of source files.
- o Redistribution in binary form must provide a disclaimer that states that the software is based in part of the work of the FreeType Team, in the distribution documentation. We also encourage you to put an URL to the FreeType web page in your documentation, though this isn't mandatory.

These conditions apply to any software derived from or based on the FreeType Project, not just the unmodified files. If you use our work, you must acknowledge us. However, no fee need be paid to us.

3. Advertising

Neither the FreeType authors and contributors nor you shall use the name of the other for commercial, advertising, or promotional purposes without specific prior written permission.

We suggest, but do not require, that you use one or more of the following phrases to refer to this software in your documentation or advertising materials: `FreeType Project', `FreeType Engine', `FreeType library', or `FreeType Distribution'.

As you have not signed this license, you are not required to accept it. However, as the FreeType Project is copyrighted material, only this license, or another one contracted with the authors, grants you the right to use, distribute, and modify it. Therefore, by using, distributing, or modifying the FreeType Project, you indicate that you understand and accept all the terms of this license.

4. Contacts

There are two mailing lists related to FreeType:

o freetype@nongnu.org

Discusses general use and applications of FreeType, as well as future and wanted additions to the library and distribution.

If you are looking for support, start in this list if you haven't found anything to help you in the documentation.

o freetype-devel@nongnu.org

Discusses bugs, as well as engine internals, design issues, specific licenses, porting, etc.

Our home page can be found at

<http://www.freetype.org>

GPL v2

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be

distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein.

You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively

convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may

consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

#####

--- fthash.c and fthash.h are covered by the following notices ---

/*

* Copyright 2000 Computing Research Labs, New Mexico State University

* Copyright 2001-2015

* Francesco Zappa Nardelli

*

* Permission is hereby granted, free of charge, to any person obtaining a

* copy of this software and associated documentation files (the "Software"),

* to deal in the Software without restriction, including without limitation

* the rights to use, copy, modify, merge, publish, distribute, sublicense,

* and/or sell copies of the Software, and to permit persons to whom the

* Software is furnished to do so, subject to the following conditions:

*

* The above copyright notice and this permission notice shall be included in

* all copies or substantial portions of the Software.

*

* THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

* IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,

* FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL

* THE COMPUTING RESEARCH LAB OR NEW MEXICO STATE UNIVERSITY BE LIABLE FOR ANY

* CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT

* OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR

* THE USE OR OTHER DEALINGS IN THE SOFTWARE.

*/

/******

*

* This file is based on code from bdf.c,v 1.22 2000/03/16 20:08:50

*

* taken from Mark Leisher's xmbdfed package

*

*/

#####

--- FreeType 2 PSaux module is covered by the following notices ---

Copyright 2006-2014 Adobe Systems Incorporated.

This software, and all works of authorship, whether in source or

object code form as indicated by the copyright notice(s) included herein (collectively, the "Work") is made available, and may only be used, modified, and distributed under the FreeType Project License, LICENSE.TXT. Additionally, subject to the terms and conditions of the FreeType Project License, each contributor to the Work hereby grants to any individual or legal entity exercising permissions granted by the FreeType Project License and this section (hereafter, "You" or "Your") a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such contributor that are necessarily infringed by their contribution(s) alone or by combination of their contribution(s) with the Work to which such contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

By using, modifying, or distributing the Work you indicate that you have read and understood the terms and conditions of the FreeType Project License as well as those provided in this section, and you accept them fully.

#####

...

Apache Commons Byte Code Engineering Library (BCEL) Version 6.4.1

Apache Commons BCEL Notice

<pre>

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                    ==
== in this case for the Apache Commons BCEL distribution. ==
=====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

</pre>

Apache 2.0 License

<pre>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including

the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each

Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

</pre>

```
## Mesa 3-D Graphics Library v21.0.3
```

```
### Mesa License
```

```
...
```

```
Copyright (C) 1999-2007 Brian Paul All Rights Reserved.
```

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Attention, Contributors

When contributing to the Mesa project you must agree to the licensing terms of the component to which you're contributing.

The following section lists the primary components of the Mesa distribution and their respective licenses.

Mesa Component Licenses

Component	Location	License

Main Mesa code src/mesa/ MIT
Device drivers src/mesa/drivers/* MIT, generally

Gallium code src/gallium/ MIT

Ext headers GL/gltext.h Khronos
GL/gltext.h Khronos
GL/wgltext.h Khronos
KHR/khrplatform.h Khronos

include/GL/gl.h :

Mesa 3-D graphics library

Copyright (C) 1999-2006 Brian Paul All Rights Reserved.

Copyright (C) 2009 VMware, Inc. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

include/GL/gltext.h
include/GL/gltext.h
include/GL/wgltext.h :

Copyright (c) 2013 - 2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

include/KHR/khrplatform.h :

Copyright (c) 2008 - 2018 The Khronos Group Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and/or associated documentation files (the "Materials"), to deal in the Materials without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Materials, and to permit persons to whom the Materials are furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Materials.

THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.

...

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Little Color Management System (LCMS) v2.12

LCMS License

<pre>

Little Color Management System
Copyright (c) 1998-2020 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

ASM Bytecode Manipulation Framework v6.0

ASM License

<pre>

Copyright (c) 2000-2011 France Tlcom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

xml-commons/java/external/LICENSE.sax.txt \$Id: LICENSE.sax.txt 225954 2002-01-31 23:26:48Z curcuru \$

This license came from: <http://www.megginson.com/SAX/copying.html>

However please note future versions of SAX may be covered

under <http://saxproject.org/?selected=pd>

This page is now out of date -- see the new SAX site at

<http://www.saxproject.org/> for more up-to-date

releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The Unicode Standard, Unicode Character Database, Version 10.0.0

Unicode Character Database

...

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories
<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,
<http://www.unicode.org/cldr/data/>,
<http://source.icu-project.org/repos/icu/>, and
<http://www.unicode.org/utility/trac/browser/>.

Unicode Data Files do not include PDF online code charts under the
directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard
or under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,
<http://www.unicode.org/cldr/data/>,
<http://source.icu-project.org/repos/icu/>, and
<http://www.unicode.org/utility/trac/browser/>.

NOTICE TO USER: Carefully read the following legal agreement.

BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S
DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),
YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE

TERMS AND CONDITIONS OF THIS AGREEMENT.

IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.

Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

...

Mozilla Elliptic Curve Cryptography (ECC)

Mozilla ECC Notice

This notice is provided with respect to Elliptic Curve Cryptography, which is included with JRE, JDK, and OpenJDK.

You are receiving a

[copy](<http://hg.openjdk.java.net/jdk9/jdk9/jdk/file/tip/src/jdk.crypto.ec/share/native/libsunec/impl>)

of the Elliptic Curve Cryptography library in source form with the JDK and OpenJDK source distributions, and as object code in the JRE & JDK runtimes.

<pre>

In the case of the JRE & JDK runtimes, the terms of the Oracle license do NOT apply to the Elliptic Curve Cryptography library; it is licensed under the following license, separately from Oracle's JDK & JRE. If you do not wish to install the Elliptic Curve Cryptography library, you may delete the Elliptic Curve Cryptography library:

- On Solaris and Linux systems: delete \$(JAVA_HOME)/lib/libsunec.so
- On Mac OSX systems: delete \$(JAVA_HOME)/lib/libsunec.dylib
- On Windows systems: delete \$(JAVA_HOME)\bin\sunec.dll

</pre>

Written Offer for Source Code

<pre>

For third party technology that you receive from Oracle in binary form which is licensed under an open source license that gives you the right to receive the source code for that binary, you can obtain a copy of the applicable source code from this page:

<http://hg.openjdk.java.net/jdk9/jdk9/jdk/file/tip/src/jdk.crypto.ec/share/native/libsunec/impl>

If the source code for the technology was not provided to you with the binary, you can also receive a copy of the source code on physical media by submitting a written request to:

Oracle America, Inc.
Attn: Associate General Counsel,
Development and Engineering Legal
500 Oracle Parkway, 10th Floor
Redwood Shores, CA 94065

Or, you may send an email to Oracle using the form at:

<http://www.oracle.com/goto/opensourcecode/request>

Your request should include:

- The name of the component or binary file(s) for which you are requesting the source code
- The name and version number of the Oracle product containing the binary
- The date you received the Oracle product
- Your name

- Your company name (if applicable)
- Your return mailing address and email and
- A telephone number in the event we need to reach you.

We may charge you a fee to cover the cost of physical media and processing. Your request must be sent (i) within three (3) years of the date you received the Oracle product that included the component or binary file(s) that are the subject of your request, or (ii) in the case of code licensed under the GPL v3, for as long as Oracle offers spare parts or customer support for that product model.

</pre>

LGPL 2.1

<pre>

GNU LESSER GENERAL PUBLIC LICENSE Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts
as the successor of the GNU Library Public License, version 2, hence
the version number 2.1.]

Preamble

The licenses for most software are designed to take away your
freedom to share and change it. By contrast, the GNU General Public
Licenses are intended to guarantee your freedom to share and change
free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some
specially designated software packages--typically libraries--of the
Free Software Foundation and other authors who decide to use it. You
can use it too, but we suggest you first think carefully about whether
this license or the ordinary General Public License is the better
strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use,
not price. Our General Public Licenses are designed to make sure that
you have the freedom to distribute copies of free software (and charge
for this service if you wish); that you receive source code or can get

it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with

the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the

Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the

Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time.

Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest

possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990

Ty Coon, President of Vice

That's all there is to it!

</pre>

xwd v1.0.7

xwd utility

<pre>

This is the copyright for the files in src/java.desktop/unix/native/libawt_xawt:

list.h, multiVis.h, wsutils.h, list.c, multiVis.c

Copyright 1994 Hewlett-Packard Co.

Copyright 1996, 1998 The Open Group

Permission to use, copy, modify, distribute, and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation.

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE OPEN GROUP BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of The Open Group shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization from The Open Group.

</pre>

Copyright (c) 2009, Orbitz World Wide

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the Orbitz World Wide nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT

OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache HttpClient Mime
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

The project util-linux doesn't use the same license for all of the code.
There is code under:

- * GPLv3+ (GNU General Public License version 3, or any later version)
- * GPLv2+ (GNU General Public License version 2, or any later version)
- * GPLv2 (GNU General Public License version 2)
- * LGPLv2+ (GNU Lesser General Public License v2 (or 2.1) or any later version)
- * BSD with advertising
- * Public Domain

Please, check the source code for more details. A license is usually at the start of each source file.

The /COPYING file (GPLv2+) is the default license for code without an explicitly defined license.

```
## CUP Parser Generator for Java v 0.11b
```

```
### CUP Parser Generator License
```

```
<pre>
```

Copyright 1996-2015 by Scott Hudson, Frank Flannery, C. Scott Ananian, Michael Petter

Permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both the copyright notice and this permission notice and warranty disclaimer appear in supporting documentation, and that the names of the authors or their employers not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission.

The authors and their employers disclaim all warranties with regard to this software, including all implied warranties of merchantability and fitness. In no event shall the authors or their employers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of this software.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate

as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify

the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include

the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Santuario v2.2.1

Apache Santuario Notice

<pre>

Apache Santuario - XML Security for Java
Copyright 1999-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

It was originally based on software copyright (c) 2001, Institute for Data Communications Systems, <<http://www.nue.et-inf.uni-siegen.de/>>.

The development of this software was partly funded by the European Commission in the <WebSig> project in the ISIS Programme.

</pre>

Apache 2.0 License

<pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

</pre>

```
## Harfbuzz v2.3.1
```

```
### Harfbuzz License
```

```
http://cgit.freedesktop.org/harfbuzz/tree/COPYING
```

<pre>

HarfBuzz is licensed under the so-called "Old MIT" license. Details follow. For parts of HarfBuzz that are licensed under different licenses see individual files names COPYING in subdirectories where applicable.

Copyright 2010,2011,2012 Google, Inc.
Copyright 2012 Mozilla Foundation
Copyright 2011 Codethink Limited
Copyright 2008,2010 Nokia Corporation and/or its subsidiary(-ies)
Copyright 2009 Keith Stribley
Copyright 2009 Martin Hosken and SIL International
Copyright 2007 Chris Wilson
Copyright 2006 Behdad Esfahbod
Copyright 2005 David Turner
Copyright 2004,2007,2008,2009,2010 Red Hat, Inc.
Copyright 1998-2004 David Turner and Werner Lemberg

For full copyright notices consult the individual files in the package.

Permission is hereby granted, without written agreement and without license or royalty fees, to use, copy, modify, and distribute this software and its documentation for any purpose, provided that the above copyright notice and the following two paragraphs appear in all copies of this software.

IN NO EVENT SHALL THE COPYRIGHT HOLDER BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE COPYRIGHT HOLDER HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE COPYRIGHT HOLDER SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN "AS IS" BASIS, AND THE COPYRIGHT HOLDER HAS NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

All source code, except for one section, is licensed as above. The one exception is licensed with a slightly different MIT variant:
The contents of this directory are licensed under the following terms:

Copyright (C) 2012 Grigori Goronzy <greg@kinoho.net>

Permission to use, copy, modify, and/or distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation

source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

APACHE COMMONS MATH DERIVATIVE WORKS:

The Apache commons-math library includes a number of subcomponents whose implementation is derived from original sources written in C or Fortran. License terms of the original sources are reproduced below.

=====
For the lmdcr, lmpar and qrsolv Fortran routine from minpack and translated in the LevenbergMarquardtOptimizer class in package org.apache.commons.math.optimization.general
Original source copyright and license statement:

Minpack Copyright Notice (1999) University of Chicago. All rights reserved

Redistribution and use in source and binary forms, with or

without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by the University of Chicago, as Operator of Argonne National Laboratory.

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. WARRANTY DISCLAIMER. THE SOFTWARE IS SUPPLIED "AS IS" WITHOUT WARRANTY OF ANY KIND. THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, AND THEIR EMPLOYEES: (1) DISCLAIM ANY WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE OR NON-INFRINGEMENT, (2) DO NOT ASSUME ANY LEGAL LIABILITY OR RESPONSIBILITY FOR THE ACCURACY, COMPLETENESS, OR USEFULNESS OF THE SOFTWARE, (3) DO NOT REPRESENT THAT USE OF THE SOFTWARE WOULD NOT INFRINGE PRIVATELY OWNED RIGHTS, (4) DO NOT WARRANT THAT THE SOFTWARE WILL FUNCTION UNINTERRUPTED, THAT IT IS ERROR-FREE OR THAT ANY ERRORS WILL BE CORRECTED.

5. LIMITATION OF LIABILITY. IN NO EVENT WILL THE COPYRIGHT HOLDER, THE UNITED STATES, THE UNITED STATES DEPARTMENT OF ENERGY, OR THEIR EMPLOYEES: BE LIABLE FOR ANY INDIRECT, INCIDENTAL, CONSEQUENTIAL, SPECIAL OR PUNITIVE DAMAGES OF ANY KIND OR NATURE, INCLUDING BUT NOT LIMITED TO LOSS OF PROFITS OR LOSS OF DATA, FOR ANY REASON WHATSOEVER, WHETHER SUCH LIABILITY IS ASSERTED ON THE BASIS OF CONTRACT, TORT (INCLUDING NEGLIGENCE OR STRICT LIABILITY), OR OTHERWISE, EVEN IF ANY OF SAID PARTIES HAS BEEN WARNED OF THE POSSIBILITY OF SUCH LOSS OR DAMAGES.

=====
Copyright and license statement for the odex Fortran routine developed by
E. Hairer and G. Wanner and translated in GraggBulirschStoerIntegrator class
in package org.apache.commons.math.ode.nonstiff:

Copyright (c) 2004, Ernst Hairer

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED
TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A
PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR
CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====
Copyright and license statement for the original lapack fortran routines
translated in EigenDecompositionImpl class in package
org.apache.commons.math.linear:

Copyright (c) 1992-2008 The University of Tennessee. All rights reserved.

\$COPYRIGHT\$

Additional copyrights may follow

\$HEADERS\$

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer listed in this license in the documentation and/or other materials provided with the distribution.
- Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

Copyright and license statement for the original Mersenne twister C routines translated in MersenneTwister class in package org.apache.commons.math.random:

Copyright (C) 1997 - 2002, Makoto Matsumoto and Takuji Nishimura, All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of its contributors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache Log4j API

Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache HttpCore

Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Thai Dictionary

Thai Dictionary License

<pre>

Copyright (C) 1982 The Royal Institute, Thai Royal Government.

Copyright (C) 1998 National Electronics and Computer Technology Center,
National Science and Technology Development Agency,
Ministry of Science Technology and Environment,
Thai Royal Government.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE

AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

Apache Log4j Core

Copyright 1999-2012 Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

ResolverUtil.java

Copyright 2005-2006 Tim Fennell

Apache HttpCore NIO

Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

ADDITIONAL INFORMATION ABOUT LICENSING

Certain files distributed by Oracle America, Inc. and/or its affiliates are subject to the following clarification and special exception to the GPLv2, based on the GNU Project exception for its Classpath libraries, known as the GNU Classpath Exception.

Note that Oracle includes multiple, independent programs in this software package. Some of those programs are provided under licenses deemed incompatible with the GPLv2 by the Free Software Foundation and others. For example, the package includes programs licensed under the Apache License, Version 2.0 and may include FreeType. Such programs are licensed to you under their original licenses.

Oracle facilitates your further distribution of this package by adding the Classpath Exception to the necessary parts of its GPLv2 code, which permits you to use that code in combination with other independent modules not licensed under the GPLv2. However, note that this would not permit you to commingle code under an incompatible license with Oracle's GPLv2 licensed code by, for example, cutting and pasting such code into a file also containing Oracle's GPLv2 licensed code and then distributing the result.

Additionally, if you were to remove the Classpath Exception from any of the files to which it applies and distribute the result, you would likely be required to license some or all of the other code in that distribution under the GPLv2 as well, and since the GPLv2 is incompatible with the license terms of some items included in the distribution by Oracle, removing the Classpath Exception could therefore effectively compromise your ability to further

distribute the package.

Failing to distribute notices associated with some files may also create unexpected legal consequences.

Proceed with caution and we recommend that you obtain the advice of a lawyer skilled in open source matters before removing the Classpath Exception or making modifications to this package which may subsequently be redistributed and/or involve the use of third party software.

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers.

Licensing

Jackson 2.x core and extension components are licensed under Apache License 2.0 To find the details that apply to this artifact see the accompanying LICENSE file.

Credits

A list of contributors may be found from CREDITS(-2.x) file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

OASIS PKCS #11 Cryptographic Token Interface v3.0

OASIS PKCS #11 Cryptographic Token Interface License

<pre>

Copyright OASIS Open 2020. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the "OASIS IPR Policy"). The full Policy may be found at the OASIS website: [<http://www.oasis-open.org/policies-guidelines/ipr>]

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set

forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. OASIS AND ITS MEMBERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THIS DOCUMENT OR ANY PART THEREOF.

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

</pre>

PLEASE READ THIS NOTICE CAREFULLY

This AppDynamics PHP-agent uses third-party library ZeroMQ v3.2.5.

The above mentioned third-party library has been modified to function as per the agent requirement. The original source code can be found here: <http://download.zeromq.org/>.

International Components for Unicode (ICU4J) v60.2

ICU4J License

...

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

Unicode Data Files do not include PDF online code charts under the directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard or under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

NOTICE TO USER: Carefully read the following legal agreement.

BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"),

YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT.

IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.

Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation

(the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

...

Cryptix AES v3.2.0

Cryptix General License

<pre>

Cryptix General License

Copyright (c) 1995-2005 The Cryptix Foundation Limited.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE CRYPTIX FOUNDATION LIMITED AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE CRYPTIX FOUNDATION LIMITED OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

JLine v3.14.0

JLine License

<pre>

Copyright (c) 2002-2018, the original author or authors.
All rights reserved.

<https://opensource.org/licenses/BSD-3-Clause>

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of JLine nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,

PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

</pre>

JRuby Joni v2.1.16

MIT License

<pre>

Copyright (c) 2017 JRuby Team

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

</pre>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 1999-2005 The Apache Software Foundation

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

GIFLIB v5.2.1

GIFLIB License

...

The GIFLIB distribution is Copyright (c) 1997 Eric S. Raymond

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

<https://sourceforge.net/p/giflib/code/ci/master/tree/openbsd-reallocarray.c>

Copyright (c) 2008 Otto Moerbeek <otto@drijf.net>

SPDX-License-Identifier: MIT

GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program--to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not

price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official

standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose

of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- a) The work must carry prominent notices stating that you modified

it, and giving a relevant date.

b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".

c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.

d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

a) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.

b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the

product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.

c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.

d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.

e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods,

procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work,

for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- a) Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
- b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
- c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
- d) Limiting the use for publicity purposes of names of licensors or authors of the material; or
- e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
- f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an

organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means,

then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

```
<program> Copyright (C) <year> <name of author>
This program comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

```
## libpng v1.6.37
```

```
### libpng License
<pre>
```

COPYRIGHT NOTICE, DISCLAIMER, and LICENSE

```
=====
```

PNG Reference Library License version 2

```
-----
```

- * Copyright (c) 1995-2019 The PNG Reference Library Authors.
- * Copyright (c) 2018-2019 Cosmin Truta.
- * Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson.
- * Copyright (c) 1996-1997 Andreas Dilger.
- * Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

The software is supplied "as is", without warranty of any kind, express or implied, including, without limitation, the warranties of merchantability, fitness for a particular purpose, title, and

non-infringement. In no event shall the Copyright owners, or anyone distributing the software, be liable for any damages or other liability, whether in contract, tort or otherwise, arising from, out of, or in connection with the software, or the use or other dealings in the software, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this software, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated, but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

PNG Reference Library License version 1 (for libpng 0.5 through 1.6.35)

libpng versions 1.0.7, July 1, 2000, through 1.6.35, July 15, 2018 are Copyright (c) 2000-2002, 2004, 2006-2018 Glenn Randers-Pehrson, are derived from libpng-1.0.6, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Mans Rullgard
Cosmin Truta
Gilles Vollant
James Yu
Mandar Sahastrabudhe
Google Inc.
Vadim Barkov

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire

risk of satisfactory quality, performance, accuracy, and effort is with the user.

Some files in the "contrib" directory and some configure-generated files that are distributed with libpng have other copyright owners, and are released under other open source licenses.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998-2000 Glenn Randers-Pehrson, are derived from libpng-0.96, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996-1997 Andreas Dilger, are derived from libpng-0.88, and are distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

Some files in the "scripts" directory have other copyright owners, but are released under this license.

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995-1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of

merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

</pre>

Apache Commons Logging
Copyright 2003-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache Log4j Commons Logging Bridge
Copyright 1999-1969 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Unicode Common Local Data Repository (CLDR) v33

CLDR License

...

UNICODE, INC. LICENSE AGREEMENT - DATA FILES AND SOFTWARE

Unicode Data Files include all data files under the directories
<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,
<http://www.unicode.org/cldr/data/>,
<http://source.icu-project.org/repos/icu/>, and
<http://www.unicode.org/utility/trac/browser/>.

Unicode Data Files do not include PDF online code charts under the directory <http://www.unicode.org/Public/>.

Software includes any source code published in the Unicode Standard or under the directories

<http://www.unicode.org/Public/>, <http://www.unicode.org/reports/>,

<http://www.unicode.org/cldr/data/>,

<http://source.icu-project.org/repos/icu/>, and

<http://www.unicode.org/utility/trac/browser/>.

NOTICE TO USER: Carefully read the following legal agreement.

BY DOWNLOADING, INSTALLING, COPYING OR OTHERWISE USING UNICODE INC.'S DATA FILES ("DATA FILES"), AND/OR SOFTWARE ("SOFTWARE"), YOU UNEQUIVOCALLY ACCEPT, AND AGREE TO BE BOUND BY, ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE, DO NOT DOWNLOAD, INSTALL, COPY, DISTRIBUTE OR USE THE DATA FILES OR SOFTWARE.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2018 Unicode, Inc. All rights reserved.

Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that either (a) this copyright and permission notice appear with all copies of the Data Files or Software, or (b) this copyright and permission notice appear in associated Documentation.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS.

IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

...

xml-commons/java/external/README.sax.txt \$Id: README.sax.txt 225954 2002-01-31 23:26:48Z curcuru \$

HEAR YE, HEAR YE!

All of the .java software and associated documentation about SAX in this repository are distributed freely in the public domain.

LICENSE.sax.txt covers all software and documentation from the megginson.com including the following in the xml-commons project:

xml-commons/java/external/src/org/xml/sax
and all subdirectories
xml-commons/java/external/xdocs/sax
and all subdirectories

The actual SAX classes in xml-commons came from:

<http://www.megginson.com/Software/index.html>

The original versions are tagged 'SAX-2_0-r2-prerelease'

=====
== NOTICE file corresponding to section 4(d) of the Apache License, ==
== Version 2.0, in this case for the Apache Xerces Java distribution. ==
=====

Apache Xerces Java
Copyright 1999-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Portions of this software were originally based on the following:

- software copyright (c) 1999, IBM Corporation., <http://www.ibm.com>.
- software copyright (c) 1999, Sun Microsystems., <http://www.sun.com>.
- voluntary contributions made by Paul Eng on behalf of the Apache Software Foundation that were originally developed at iClick, Inc., software copyright (c) 1999.

The copyrights in this software and any visual or audio work distributed with the software belong to Azul Systems, Inc. and those included in all other notice

files either listed in the readme file or contained in any other included notice files with this distribution. All rights are reserved. Installation of this software and any Azul software bundled with or derived from this software is licensed only in accordance with these terms.

Provided you have not received the software directly from Azul and have already agreed to the terms of a separate license agreement, by installing, using or distributing this software you, on your own behalf and on behalf of your employer or principal, agree to be bound by these terms. If you do not agree to any of these terms, you may not use, copy, transmit, distribute nor install this software.

The software is developed and owned by Azul and/or any of its affiliates, subsidiaries or respective suppliers and licensors. The software also includes certain software components and materials ("Open Source Materials") of third parties ("Third Party Licensors") licensed under certain licenses ("Open Source Licenses").

Nothing in this Agreement limits your rights under, or grants you rights that supersede the terms and conditions of any applicable Open Source Licenses. You must review the Open Source Licenses located at http://www.azulsystems.com/license/zulu_third_party_licenses.html to understand your rights under them. In the event that no such third party license agreements exist, the restrictions contained in this Agreement shall apply.

With respect to the Open Source Licenses the following shall apply:

- 1) You hereby acknowledge and agree that you will be licensing any Open Source Materials directly from the applicable Third Party Licensors including the right to use such Third Party Materials in connection with the software
- 2) In the event of any inconsistencies or conflicting provisions of the Open Source Licenses and the provisions of this Agreement, the provisions of the Open Source Licenses shall prevail.

THIS SOFTWARE IS PROVIDED BY AZUL "AS-IS" AND WITHOUT EXPRESS OR IMPLIED WARRANTIES, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL AZUL BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY OR CONSEQUENTIAL DAMAGES (INCLUDING BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICE, LOSS OF USE, DATA OR PROFITS; OR BUSINESS INTERRUPTION, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY WHETHER IN CONTRACT, STRICT LIABILITY OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally

submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache Commons Codec
Copyright 2002-2019 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java
contains test data from <http://aspell.net/test/orig/batch0.tab>.
Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

=====

The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at <http://stevemorse.org/phoneticinfo.htm> with permission from the original authors.
Original source copyright:
Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache HttpClient
Copyright 1999-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).
xml-commons/java/external/LICENSE.dom-documentation.txt \$Id: LICENSE.dom-documentation.txt 226215
2005-06-03 22:49:13Z mrglavas \$

This license came from: <http://www.w3.org/Consortium/Legal/copyright-documents-20021231>

W3C DOCUMENT LICENSE
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>

Public documents on the W3C site are provided by the copyright holders under the following license. By using and/or copying this document, or the W3C document from which this statement is linked, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to copy, and distribute the contents of this document, or the W3C document from which this statement is linked, in any medium for any purpose

and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the document, or portions thereof, that you use:

1. A link or URL to the original W3C document.
2. The pre-existing copyright notice of the original author, or if it doesn't exist, a notice (hypertext is preferred, but a textual representation is permitted) of the form: "Copyright [date-of-document] World Wide Web Consortium, (Massachusetts Institute of Technology, European Research Consortium for Informatics and Mathematics, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/2002/copyright-documents-20021231>"
3. If it exists, the STATUS of the W3C document.

When space permits, inclusion of the full text of this NOTICE should be provided. We request that authorship attribution be provided in any software, documents, or other items or products that you create pursuant to the implementation of the contents of this document, or any portion thereof.

No right to create modifications or derivatives of W3C documents is granted pursuant to this license. However, if additional requirements (documented in the Copyright FAQ) are satisfied, the right to create modifications or derivatives is sometimes granted by the W3C to individuals complying with those requirements.

THIS DOCUMENT IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THE DOCUMENT ARE SUITABLE FOR ANY PURPOSE; NOR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE DOCUMENT OR THE PERFORMANCE OR IMPLEMENTATION OF THE CONTENTS THEREOF.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to this document or its contents without specific, written prior permission. Title to copyright in this document will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on December 31 2002. This version removes the copyright ownership notice such that this license can be used with materials other than those owned by the W3C, moves information on style sheets, DTDs, and schemas to the Copyright FAQ, reflects that ERCIM is now a host of the W3C, includes references to this specific dated version of the license, and removes the ambiguous grant of "use". See the older

formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, such as the translating or annotating specifications. Other questions about this notice can be directed to site-policy@w3.org.

Joseph Reagle <site-policy@w3.org>

Last revised by Reagle \$Date: 2005-06-03 18:49:13 -0400 (Fri, 03 Jun 2005) \$
Apache Commons FileUpload
Copyright 2002-2018 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

A Classpath Exception File means any source file contained in this distribution which contains the following words in such files header: Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the LICENSE file that accompanied this code.

Azul Systems, Inc. hereby confirms that each Classpath Exception File is subject to the clarification and special exception to the GPL that is outlined in the accompanying LICENSE file (under the heading CLASSPATH EXCEPTION TO THE GPL).

1.216 jetbrains-kotlin-kotlin-stdlib-jdk8 1.6.10

1.216.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.217 kotlinox-io-jvm 0.1.16

1.217.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.218 jakarta-activation 1.1

1.218.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and
- (b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).
- (c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.
- (d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

- (a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and
- (b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).
- (c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.
- (d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License.

You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48C.F.R.2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48C.F.R.12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE

(CDDL)

The GlassFish code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

1.219 fabric8-:::-kubernetes-model-:::- discovery 4.10.3

1.219.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.220 micronaut-azure 3.2.3

1.220.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.221 dropwizard-logging-support 2.0.18

1.221.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.222 jvm-integration-for-metrics 4.1.17

1.222.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.codahale.metrics.jvm
Build-Jdk-Spec: 1.8
Bundle-Description: A set of classes which allow you to monitor critical aspects of your Java Virtual Machine using Metrics.
Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>
Bundle-ManifestVersion: 2
Bundle-Name: JVM Integration for Metrics
Bundle-SymbolicName: io.dropwizard.metrics.jvm
Bundle-Version: 4.1.17
Created-By: Apache Maven Bundle Plugin
Export-Package: com.codahale.metrics.jvm;uses:="com.codahale.metrics,javax.management";version="4.1.17"
Implementation-Title: JVM Integration for Metrics
Implementation-Version: 4.1.17
Import-Package: com.codahale.metrics;version="[4.1,5)",com.sun.management;resolution:=optional,javax.management,org.slf4j;version="[1.6.0,2.0.0)"
Private-Package: com.codahale.metrics.jvm
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188601_1654735114.1485894/0/metrics-jvm-4-1-17-jar/META-INF/MANIFEST.MF

1.223 stringtemplate4 4.1

1.223.1 Available under license :

[The "BSD license"]

Copyright (c) 2011-2013 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ANTLR Project Contributors Certification of Origin and Rights

All contributors to StringTemplate v4 must formally agree to abide by this certificate of origin by signing on the bottom with their github userid, full name, email address (you can obscure your e-mail, but it must be computable by human), and date.

By signing this agreement, you are warranting and representing that you have the right to release code contributions or other content free of any obligations to third parties and are granting Terence Parr and ANTLR project contributors, henceforth referred to as The ANTLR Project, a license to incorporate it into The ANTLR Project tools (such as ANTLRWorks and StringTemplate) or related works under the BSD license. You understand that The ANTLR Project may or may not incorporate your contribution and you warrant and represent the following:

1. I am the creator of all my contributions. I am the author of all contributed work submitted and further warrant and represent that such work is my original creation and I have the right to license it to The ANTLR Project for release under the 3-clause BSD license. I hereby grant The ANTLR Project a nonexclusive, irrevocable, royalty-free, worldwide license to reproduce, distribute, prepare derivative works, and otherwise use this contribution as part of the ANTLR project, associated documentation, books, and tools at no cost to The ANTLR Project.
2. I have the right to submit. This submission does not violate the rights of any person or entity and that I have legal authority over this submission and to make this certification.
3. If I violate another's rights, liability lies with me. I agree to defend, indemnify, and hold The ANTLR Project and ANTLR users

harmless from any claim or demand, including reasonable attorney fees, made by any third party due to or arising out of my violation of these terms and conditions or my violation of the rights of another person or entity.

4. I understand and agree that this project and the contribution are public and that a record of the contribution (including all personal information I submit with it, including my sign-off) is maintained indefinitely and may be redistributed consistent with this project or the open source license indicated in the file.

I have read this agreement and do so certify by adding my signoff to the end of the following contributors list.

CONTRIBUTORS:

YYYY/MM/DD, github id, Full name, email

2012/07/12, parrt, Terence Parr, parrt@antlr.org

2012/08/13, pgelinas, Pascal Glinas, pascal.gelinas@polymtl.ca

2015/05/28, jsnyders, John Snyders, jjsnyders at rcn.com

2015/12/07, sharwell, Sam Harwell, sam@tunnelvisionlabs.com

2016/08/23, BurtHarris, Burt Harris, Burt_Harris.github@azxs.33mail.com

2016/07/18, jeff5, Jeff Allen, ja.py@farowl.co.uk

2018/11/06, drealeed , Drea Leed, drealeed2@yahoo.com

2018/11/08, leonlee, Leon Lee, blackicebird@gmail.com

1.224 kotlin-stdlib-common 1.6.10

1.224.1 Available under license :

Apache-2.0

1.225 dmfs/optional 0.3

1.225.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 dmfs GmbH

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/NullSafe.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/Absent.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/Present.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/Optional.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017 dmfs GmbH

*

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/First.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/iter/OptItor.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/decorators/Mapped.java
* /opt/cola/permits/1162865425_1649200933.51/0/optional-0-3-sources-jar/org/dmfs/optional/iter/OptIter.java

1.226 fabric8-::-kubernetes-model-::-events

4.10.3

1.226.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.227 micronaut-servlet 3.2.2

1.227.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2020 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletCookies.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletResponseFactory.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/encoders/StreamFileEncoder.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletRequestBinder.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/StreamedServletMessage.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletHttpHandler.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletBodyBinder.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletRequestAndBody.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/DefaultServletExchange.java  
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-  
jar/io/micronaut/servlet/http/ServletExchange.java
```

* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/encoders/AbstractFileEncoder.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/ServletResponseEncoder.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/encoders/FileEncoder.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/ServletHttpResponse.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/ServletBinderRegistry.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/ServletHttpRequest.java
* /opt/cola/permits/1331473542_1653512935.5710146/0/micronaut-servlet-core-3-2-2-sources-jar/io/micronaut/servlet/http/encoders/SystemFileEncoder.java

1.228 netty/transport/classes/kqueue

4.1.74.Final

1.228.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Implementation-Title: Netty/Transport/Classes/KQueue

Bundle-Description: Netty is an asynchronous event-driven network application framework for rapid development of maintainable high performance protocol servers and clients.

Automatic-Module-Name: io.netty.transport.classes.kqueue

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0>

Bundle-SymbolicName: io.netty.transport-classes-kqueue

Implementation-Version: 4.1.74.Final

Built-By: norman

Bnd-LastModified: 1644314372947

Bundle-ManifestVersion: 2

Implementation-Vendor-Id: io.netty

Bundle-DocURL: <https://netty.io/>

Bundle-Vendor: The Netty Project

Import-Package: io.netty.buffer;version="[4.1,5)",io.netty.channel,io.netty.channel.socket;version="[4.1,5)",io.netty.channel.unix;version="[4.1,5)",io.netty.util;version="[4.1,5)",io.netty.util.collection;version="[4.1,5)",io.netty.util.concurrent;version="[4.1,5)",io.netty.util.internal;version="[4.1,5)",io.netty.util.internal.logging;version="[4.1,5)",sun.nio.ch;resolution:=optional,org.eclipse.jetty.npn;version="[1,2)";resolution:=optional,org.eclipse.jetty.alpn;version="[1,2)";resolution:=optional

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.6))"

Tool: Bnd-2.4.1.201501161923

Implementation-Vendor: The Netty Project
Export-Package: io.netty.channel.kqueue;uses:="io.netty.buffer,io.netty.channel,io.netty.channel.socket,io.netty.channel.unix,io.netty.util.concurrent";version="4.1.74"
Bundle-Name: Netty/Transport/Classes/KQueue
Bundle-Version: 4.1.74.Final
Created-By: Apache Maven Bundle Plugin
Build-Jdk: 1.8.0_252
Implementation-URL: <https://netty.io/netty-transport-classes-kqueue/>

Found in path(s):

* /opt/cola/permits/1340188613_1654733179.7327774/0/netty-transport-classes-kqueue-4-1-74-final-jar/META-INF/MANIFEST.MF

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2021 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188613_1654733179.7327774/0/netty-transport-classes-kqueue-4-1-74-final-jar/META-INF/maven/io.netty/netty-transport-classes-kqueue/pom.xml

1.229 junit 4.10

1.229.1 Available under license :

No license file was found, but licenses were detected in source scan.

// Copyright 2010 Google Inc. All Rights Reserved.

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1018976869_1591372852.74/0/junit-4-10-sources-jar/org/junit/runners/model/MultipleFailureException.java

1.230 kotlinox-coroutines-jdk 1.3.2-1.3.60

1.230.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.231 commons-compress 1.21

1.231.1 Available under license :

Apache Commons Compress
Copyright 2002-2021 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

The files in the package `org.apache.commons.compress.archivers.sevenz`
were derived from the LZMA SDK, version 9.20 (C/ and CPP/7zip/),
which has been placed in the public domain:

"LZMA SDK is placed in the public domain." (<http://www.7-zip.org/sdk.html>)

The test file `lbzip2_32767.bz2` has been copied from `libbzip2`'s source
repository:

This program, "bzip2", the associated library "libbzip2", and all
documentation, are copyright (C) 1996-2019 Julian R Seward. All
rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

2. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
3. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
4. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Julian Seward, jseward@acm.org

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.232 io-grpc-grpc-stub 1.39.0

1.232.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified portion of 'OkHttp', an open source
HTTP & SPDY client for Android and Java applications, which can be obtained
at:

- * LICENSE:
 - * [okhttp/third_party/okhttp/LICENSE](http://okhttp.org/third_party/okhttp/LICENSE) (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * [okhttp/third_party/okhttp](http://okhttp.org/third_party/okhttp)

This product contains a modified portion of 'Envoy', an open source
cloud-native high-performance edge/middle/service proxy, which can be
obtained at:

- * LICENSE:
 - * [xds/third_party/envoy/LICENSE](http://xds.org/third_party/envoy/LICENSE) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/envoy/NOTICE](http://xds.org/third_party/envoy/NOTICE)
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/envoy](http://xds.org/third_party/envoy)

This product contains a modified portion of 'protoc-gen-validate (PGV)',
an open source protoc plugin to generate polyglot message validators,
which can be obtained at:

- * LICENSE:
 - * [xds/third_party/protoc-gen-validate/LICENSE](http://xds.org/third_party/protoc-gen-validate/LICENSE) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/protoc-gen-validate/NOTICE](http://xds.org/third_party/protoc-gen-validate/NOTICE)

- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * `xds/third_party/protoc-gen-validate`

This product contains a modified portion of 'udpa',
an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * `xds/third_party/udpa/LICENSE` (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * `xds/third_party/udpa`

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical
transformation or translation of a Source form, including but
not limited to compiled object code, generated documentation,
and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.233 jersey-ext-entity-filtering 2.31

1.233.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made

available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology,

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely

in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available

under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such

interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in

accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute

so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free

programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or

(at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

```
## CLASSPATH EXCEPTION
```

Linking this library statically or dynamically with other modules is

making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.234 micronaut-test 3.1.1

1.234.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331474237_1653506865.917328/0/micronaut-test-junit5-3-1-1-sources-
jar/io/micronaut/test/extensions/junit5/annotation/MicronautTest.java
* /opt/cola/permits/1331474237_1653506865.917328/0/micronaut-test-junit5-3-1-1-sources-
jar/io/micronaut/test/extensions/junit5/MicronautJunit5Extension.java
* /opt/cola/permits/1331474237_1653506865.917328/0/micronaut-test-junit5-3-1-1-sources-
jar/io/micronaut/test/extensions/junit5/graal/MicronautJUnit5Graal.java
```


1.235 annotation-for-jdk-requirements 1.0

1.235.1 Available under license :

Found license 'Common Development and Distribution License (CDDL) 1.1' in '<name>Common Development and Distribution License</name>'

1.236 json-java 20160212

1.236.1 Available under license :

=====
Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.237 apache-commons-lang 3.11

1.237.1 Available under license :

Apache Commons Lang
Copyright 2001-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<https://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.238 junit-jupiter-junit-jupiter-api 5.8.2

1.238.1 Available under license :

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **ii)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or

subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)*** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

- * **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
- * **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
- * **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
- * **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- * **a)*** it must be made available under this Agreement, or if the Program *(i)* is combined with other material in a separate file or files made available under a Secondary License, and *(ii)* the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- * **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual

property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

1.239 snappy-java 1.1.8.1

1.239.1 Available under license :

This product includes software developed by Google

Snappy: <http://code.google.com/p/snappy/> (New BSD License)

This product includes software developed by Apache

PureJavaCrc32C from apache-hadoop-common <http://hadoop.apache.org/>

(Apache 2.0 license)

This library containd statically linked libstdc++. This inclusion is allowed by

"GCC RUnTime Library Exception"

<http://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html>

== Contributors ==

* Tatu Saloranta

* Providing benchmark suite

* Alec Wysoker

* Performance and memory usage improvement

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.240 jetty-java-based-http-1-x-http-2-servlet-websocket-server 11.0.6

1.240.1 Available under license :

Notices for Eclipse Jetty

=====

This content is produced and maintained by the Eclipse Jetty project.

Project home: <https://www.eclipse.org/jetty/>

Trademarks

Eclipse Jetty, and Jetty are trademarks of the Eclipse Foundation.

Copyright

All contributions are the property of the respective authors or of entities to which copyright has been assigned by the authors (eg. employer).

Declared Project Licenses

This artifacts of this project are made available under the terms of:

* the Eclipse Public License v2.0

<https://www.eclipse.org/legal/epl-2.0>

SPDX-License-Identifier: EPL-2.0

or

* the Apache License, Version 2.0

<https://www.apache.org/licenses/LICENSE-2.0>

SPDX-License-Identifier: Apache-2.0

The following dependencies are EPL.

* org.eclipse.jetty.orbit:org.eclipse.jdt.core

The following dependencies are EPL and ASL2.

- * org.eclipse.jetty.orbit:javax.security.auth.message

The following dependencies are EPL and CDDL 1.0.

- * org.eclipse.jetty.orbit:javax.mail.glassfish

The following dependencies are CDDL + GPLv2 with classpath exception.

<https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html>

- * jakarta.servlet:jakarta.servlet-api
- * javax.annotation:javax.annotation-api
- * javax.transaction:javax.transaction-api
- * javax.websocket:javax.websocket-api

The following dependencies are licensed by the OW2 Foundation according to the terms of <http://asm.ow2.org/license.html>

- * org.ow2.asm:asm-commons
- * org.ow2.asm:asm

The following dependencies are ASL2 licensed.

- * org.apache.taglibs:taglibs-standard-spec
- * org.apache.taglibs:taglibs-standard-impl

The following dependencies are ASL2 licensed. Based on selected classes from following Apache Tomcat jars, all ASL2 licensed.

- * org.mortbay.jasper:apache-jsp
- * org.apache.tomcat:tomcat-jasper
- * org.apache.tomcat:tomcat-juli
- * org.apache.tomcat:tomcat-jsp-api
- * org.apache.tomcat:tomcat-el-api
- * org.apache.tomcat:tomcat-jasper-el
- * org.apache.tomcat:tomcat-api
- * org.apache.tomcat:tomcat-util-scan
- * org.apache.tomcat:tomcat-util
- * org.mortbay.jasper:apache-el
- * org.apache.tomcat:tomcat-jasper-el
- * org.apache.tomcat:tomcat-el-api

The following artifacts are CDDL + GPLv2 with classpath exception.

<https://glassfish.dev.java.net/nonav/public/CDDL+GPL.html>

- * org.eclipse.jetty.toolchain:jetty-schemas

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

The UnixCrypt.java code implements the one way cryptography used by Unix systems for simple password protection. Copyright 1996 Aki Yoshida, modified April 2001 by Iris Van den Broeke, Daniel Deville.

Permission to use, copy, modify and distribute UnixCrypt for non-commercial or commercial purposes and without fee is granted provided that the copyright notice appears in all copies.

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the

licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial

product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's

rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to

look for such a notice.

You may add additional accurate notices of copyright ownership.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

SPDX-License-Identifier: EPL-2.0 OR Apache-2.0

1.241 snake-yaml 1.30

1.241.1 Available under license :

Found license 'GNU Lesser General Public License' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, <http://www.eclipse.org/legal> // LGPL, GNU Lesser General Public License, V2.1 or later, <http://www.gnu.org/licenses/lgpl.html> // GPL, GNU General Public License, V2 or later, <http://www.gnu.org/licenses/gpl.html> // AL, Apache License, V2.0 or later, <http://www.apache.org/licenses> // BSD, BSD License, <http://www.opensource.org/licenses/bsd-license.php> * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

Found license 'Eclipse Public License 1.0' in '// This module is multi-licensed and may be used under the terms // EPL, Eclipse Public License, V1.0 or later, <http://www.eclipse.org/legal> // LGPL, GNU Lesser General Public License, V2.1 or later, <http://www.gnu.org/licenses/lgpl.html> // GPL, GNU General Public License, V2 or later, <http://www.gnu.org/licenses/gpl.html> // AL, Apache License, V2.0 or later, <http://www.apache.org/licenses> // BSD, BSD License, <http://www.opensource.org/licenses/bsd-license.php> * Multi-licensed: EPL / LGPL / GPL / AL / BSD.'

1.242 protobuf-java-format 1.2

1.242.1 Available under license :

Copyright (c) 2009, Orbitz World Wide
All rights reserved.

Redistribution and use in source and binary forms, with or without modification,
are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice,
this list of conditions and the following disclaimer.

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the Orbitz World Wide nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2009, Orbitz LLC
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the Orbitz LLC nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.243 jersey's-jersey 3.0.2

1.243.1 Available under license :

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Notice for Jersey Core Common module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Google Guava Version 18.0

* License: Apache License, 2.0

* Copyright: (C) 2009 The Guava Authors

JSR-166 Extension - JEP 266

* License: Creative Commons 1.0 (CC0)

* No copyright

* Written by Doug Lea with assistance from members of JCP JSR-166

* Expert Group and released to the public domain, as explained at

* <http://creativecommons.org/publicdomain/zero/1.0/>

Notice for Jersey Core Server module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made

available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

org.objectweb.asm Version 9.0

* License: Modified BSD (<http://asm.objectweb.org/license.html>)

* Copyright: (c) 2000-2011 INRIA, France Telecom. All rights reserved.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

/*

* Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.

*

* This program and the accompanying materials are made available under the

* terms of the Eclipse Public License v. 2.0, which is available at

* <http://www.eclipse.org/legal/epl-2.0>.

*

* This Source Code may also be made available under the following Secondary

* Licenses when the conditions for such availability set forth in the

* Eclipse Public License v. 2.0 are satisfied: GNU General Public License,

* version 2 with the GNU Classpath Exception, which is available at

* <https://www.gnu.org/software/classpath/license.html>.

*

* SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

*/

Notice for Jersey Bean Validation module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Hibernate Validator CDI, 7.0.0.Final

* License: Apache License, 2.0

* Project: <https://beanvalidation.org/>

* Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Notice for Jersey Json Jackson module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Jackson JAX-RS Providers version 2.11.3

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at <https://www.eclipse.org/org/documents/edl-v10.php>.

SPDX-License-Identifier: BSD-3-Clause

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey/examples>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Coyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 3.0.0

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

CDI API Version 3.0

- * License: Apache License, 2.0
- * Project: <http://www.seamframework.org/Weld>
- * Copyright 2010, Red Hat, Inc., and individual contributors by the @authors tag.

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

jakarta.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.11.3

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license

- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 9.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>
- # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any

exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

- a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and
- b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:
 - i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and
 - iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published,

Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free

Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either

verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR

OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this

exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

/*

* Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.

*

* This program and the accompanying materials are made available under the

* terms of the Eclipse Distribution License v. 1.0, which is available at

* <http://www.eclipse.org/org/documents/edl-v10.php>.

*

* SPDX-License-Identifier: BSD-3-Clause

*/

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

* License MIT (<http://www.opensource.org/licenses/mit-license.php>)

* Project: <http://angularjs.org>

* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

* License: all the source code provided by AOP Alliance is Public Domain.

* Project: <http://aopalliance.sourceforge.net>

* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 3.0.0

* License: Apache License, 2.0

* Project: <http://beanvalidation.org/1.1/>

* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors

* by the @authors tag.

Hibernate Validator CDI, 7.0.0.Final

* License: Apache License, 2.0

* Project: <https://beanvalidation.org/>

* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7

* License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

* Project: <http://getbootstrap.com>

* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

* License: Apache License, 2.0

* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

* License: Apache License, 2.0

* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

* License: Apache License, 2.0

* Project: <http://www.javassist.org/>

* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.11.3

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

* License: jquery.org/license

* Project: jquery.org

* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

* License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> &

<http://www.gnu.org/licenses/gpl.html>)

* Project: <http://www.pasella.it/projects/jQuery/barcode>

* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

* License: CC0

* No copyright

* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

* License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)

* Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>

* Copyright: Eric Rowell

org.objectweb.asm Version 9.0

* License: Modified BSD (<http://asm.objectweb.org/license.html>)

* Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

* License: Apache License, 2.0

* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

1.244 kotlinx-coroutines-bom 1.5.1

1.244.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,

and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the

Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory,

whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2000-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License (MIT)

Copyright (c) 2016 Parker Moore

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
== NOTICE file corresponding to the section 4 d of ==
== the Apache License, Version 2.0, ==
== in this case for the kotlin.coroutines library. ==
=====

kotlin.coroutines library.
Copyright 2016-2019 JetBrains s.r.o and respective authors and developers

1.245 jetbrains-kotlin-kotlin-stdlib-jdk7 1.6.10

1.245.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.246 javax-annotation-api 1.3.2

1.246.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the

terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the

Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software

against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and

that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are

different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception

statement from your version.

1.247 kqueue 4.1.74.Final

1.247.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2016 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-jar/META-INF/maven/io.netty/netty-transport-native-kqueue/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-jar/netty_kqueue_eventarray.h


```
* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-  
jar/netty_kqueue_native.c  
* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-  
jar/netty_kqueue_eventarray.c  
* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-  
jar/netty_kqueue_bsdsocket.h  
* /opt/cola/permits/1340188523_1654650491.343734/0/netty-transport-native-kqueue-4-1-74-final-sources-2-  
jar/netty_kqueue_bsdsocket.c
```

1.248 ktor-network-tls 1.2.6

1.248.1 Available under license :

```
/*
```

```
* Copyright 2014-2019 JetBrains s.r.o and contributors. Use of this source code is governed by the Apache 2.0  
license.
```

```
*/
```

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.249 feign-slf4j 8.18.0

1.249.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274703681_1645234024.72/0/feign-slf4j-8-18-0-sources-jar/feign/slf4j/Slf4jLogger.java

1.250 perfmark-perfmark-api 0.23.0

1.250.1 Available under license :

Copyright 2019 Google LLC

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'Catapult', an open source Trace Event viewer for Chrome, Linux, and Android applications, which can be obtained at:

* LICENSE:

* [traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/catapult/LICENSE](#) (New BSD License)

* HOMEPAGE:

* <https://github.com/catapult-project/catapult>

This product contains a modified portion of 'Polymer', a library for Web Components, which can be obtained at:

* LICENSE:

* [traceviewer/src/main/resources/io/perfmark/traceviewer/third_party/polymer/LICENSE](#) (New BSD License)

* HOMEPAGE:

* <https://github.com/Polymer/polymer>

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of catapult nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

```
// Copyright (c) 2012 The Polymer Authors. All rights reserved.  
//  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions are  
// met:  
//  
// * Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// * Redistributions in binary form must reproduce the above  
// copyright notice, this list of conditions and the following disclaimer  
// in the documentation and/or other materials provided with the  
// distribution.  
// * Neither the name of Google Inc. nor the names of its  
// contributors may be used to endorse or promote products derived from  
// this software without specific prior written permission.  
//  
// THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS  
// "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT  
// LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR  
// A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
```

```
// OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,  
// SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT  
// LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,  
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY  
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT  
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE  
// OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.  
/*  
* Copyright 2019 Google LLC  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

1.251 jakarta-dependency-injection 2.0.1

1.251.1 Available under license :

Notices for Eclipse Jakarta Dependency Injection

This content is produced and maintained by the Eclipse Jakarta Dependency Injection project.

* Project home: <https://projects.eclipse.org/projects/cdi.batch>

Trademarks

Jakarta Dependency Injection is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Apache License, Version 2.0 which is available at <https://www.apache.org/licenses/LICENSE-2.0>.

SPDX-License-Identifier: Apache-2.0

Source Code

The project maintains the following source code repositories:

<https://github.com/eclipse-ee4j/injection-api>
<https://github.com/eclipse-ee4j/injection-spec>
<https://github.com/eclipse-ee4j/injection-tck>

Third-party Content

This project leverages the following third party content.

None

Cryptography

None

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation

source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

(except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and

may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify,

defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.252 dropwizard-jetty-support 2.0.18

1.252.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.253 throttling-appender 1.1.0

1.253.1 Available under license :

No license file was found, but licenses were detected in source scan.

<name>Apache License 2.0</name>

<url><http://www.apache.org/licenses/LICENSE-2.0.html></url>

Found in path(s):

* /opt/cola/permits/1288519811_1647855410.0/0/logback-throttling-appender-1-1-0-sources-jar/META-INF/maven/io.dropwizard.logback/logback-throttling-appender/pom.xml

1.254 kotlinox-io 0.1.16

1.254.1 Available under license :

Apache License

Version 2.0, January 2004

<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.255 io-grpc-grpc-core 1.39.0

1.255.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a modified portion of 'OkHttp', an open source HTTP & SPDY client for Android and Java applications, which can be obtained at:

- * LICENSE:
 - * [okhttp/third_party/okhttp/LICENSE](#) (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * [okhttp/third_party/okhttp](#)

This product contains a modified portion of 'Envoy', an open source cloud-native high-performance edge/middle/service proxy, which can be obtained at:

- * LICENSE:
 - * [xds/third_party/envoy/LICENSE](#) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/envoy/NOTICE](#)
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/envoy](#)

This product contains a modified portion of 'protoc-gen-validate (PGV)', an open source protoc plugin to generate polyglot message validators, which can be obtained at:

- * LICENSE:
 - * [xds/third_party/protoc-gen-validate/LICENSE](#) (Apache License 2.0)
- * NOTICE:
 - * [xds/third_party/protoc-gen-validate/NOTICE](#)
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * [xds/third_party/protoc-gen-validate](#)

This product contains a modified portion of 'udpa',

an open source universal data plane API, which can be obtained at:

* LICENSE:

* xds/third_party/udpa/LICENSE (Apache License 2.0)

* HOMEPAGE:

* <https://github.com/cncf/udpa>

* LOCATION_IN_GRPC:

* xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.256 netty/tomcatnative-[openssl---classes]

2.0.50.Final

1.256.1 Available under license :

LICENSE ISSUES

=====

The OpenSSL toolkit stays under a double license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts.

OpenSSL License

/* =====

* Copyright (c) 1998-2019 The OpenSSL Project. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the
 * distribution.
 *

* 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"
 *

* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without
 * prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *

* 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *

* 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *

* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *

* This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *

* This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 * must display the following acknowledgement:
 * "This product includes cryptographic software written by
 * Eric Young (eay@cryptsoft.com)"
 * The word 'cryptographic' can be left out if the rouines from the library
 * being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 * the apps directory (application code) you must include an acknowledgement:
 * "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
 * THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *

- * The licence and distribution terms for any publically available version or
- * derivative of this code cannot be changed. i.e. this code cannot simply be
- * copied and put under another distribution licence
- * [including the GNU Public Licence.]
- */

BoringSSL is a fork of OpenSSL. As such, large parts of it fall under OpenSSL licensing. Files that are completely new have a Google copyright and an ISC license. This license is reproduced at the bottom of this file.

Contributors to BoringSSL are required to follow the CLA rules for Chromium:
<https://cla.developers.google.com/cla>

Files in third_party/ have their own licenses, as described therein. The MIT license, for third_party/fiat, which, unlike other third_party directories, is compiled into non-test libraries, is included below.

The OpenSSL toolkit stays under a dual license, i.e. both the conditions of the OpenSSL License and the original SSLeay license apply to the toolkit. See below for the actual license texts. Actually both licenses are BSD-style Open Source licenses. In case of any license issues related to OpenSSL please contact openssl-core@openssl.org.

The following are Google-internal bug numbers where explicit permission from some authors is recorded for use of their work. (This is purely for our own record keeping.)

27287199

27287880

27287883

OpenSSL License

/* =====

* Copyright (c) 1998-2011 The OpenSSL Project. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:

*

* 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.

*

* 3. All advertising materials mentioning features or use of this
 * software must display the following acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit. (<http://www.openssl.org/>)"

*

* 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 * endorse or promote products derived from this software without

* prior written permission. For written permission, please contact
 * openssl-core@openssl.org.
 *

* 5. Products derived from this software may not be called "OpenSSL"
 * nor may "OpenSSL" appear in their names without prior written
 * permission of the OpenSSL Project.
 *

* 6. Redistributions of any form whatsoever must retain the following
 * acknowledgment:
 * "This product includes software developed by the OpenSSL Project
 * for use in the OpenSSL Toolkit (<http://www.openssl.org/>)"
 *

* THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS" AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * =====
 *

* This product includes cryptographic software written by Eric Young
 * (eay@cryptsoft.com). This product includes software written by Tim
 * Hudson (tjh@cryptsoft.com).
 *
 */

Original SSLeay License

 /* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to. The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code. The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in

* the code are not to be removed.

* If this package is used in a product, Eric Young should be given attribution

* as the author of the parts of the library used.

* This can be in the form of a textual message at program startup or

* in documentation (online or textual) provided with the package.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. All advertising materials mentioning features or use of this software

* must display the following acknowledgement:

* "This product includes cryptographic software written by

* Eric Young (eay@cryptsoft.com)"

* The word 'cryptographic' can be left out if the routines from the library

* being used are not cryptographic related :-).

* 4. If you include any Windows specific code (or a derivative thereof) from

* the apps directory (application code) you must include an acknowledgement:

* "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"

*

* THIS SOFTWARE IS PROVIDED BY ERIC YOUNG ``AS IS" AND

* ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE

* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE

* ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE

* FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL

* DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS

* OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)

* HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT

* LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY

* OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF

* SUCH DAMAGE.

*

* The licence and distribution terms for any publically available version or

* derivative of this code cannot be changed. i.e. this code cannot simply be

* copied and put under another distribution licence

* [including the GNU Public Licence.]

*/

ISC license used for completely new code in BoringSSL:

/* Copyright (c) 2015, Google Inc.

*

* Permission to use, copy, modify, and/or distribute this software for any

* purpose with or without fee is hereby granted, provided that the above

* copyright notice and this permission notice appear in all copies.

*

* THE SOFTWARE IS PROVIDED "AS IS" AND THE AUTHOR DISCLAIMS ALL WARRANTIES
* WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES OF
* MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY
* SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL DAMAGES OR ANY DAMAGES
* WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION
* OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN
* CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE. */

The code in third_party/ fiat carries the MIT license:

Copyright (c) 2015-2016 the fiat-crypto authors (see
<https://github.com/mit-plv/fiat-crypto/blob/master/AUTHORS>).

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.

Licenses for support code

Parts of the TLS test suite are under the Go license. This code is not included
in BoringSSL (i.e. libcrypto and libssl) when compiled, however, so
distributing code linked against BoringSSL does not trigger this license:

Copyright (c) 2009 The Go Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

* Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with the
distribution.

* Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived from
this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,

SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

BoringSSL uses the Chromium test infrastructure to run a continuous build, trybots etc. The scripts which manage this, and the script for generating build metadata, are under the Chromium license. Distributing code linked against BoringSSL does not trigger this license.

Copyright 2015 The Chromium Authors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Google Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The Netty Project

=====

Please visit the Netty web site for more information:

* <http://netty.io/>

Copyright 2016 The Netty Project

The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product contains a forked and modified version of Tomcat Native

- * LICENSE:
 - * [license/LICENSE.tomcat-native.txt](#) (Apache License 2.0)
- * HOMEPAGE:
 - * <http://tomcat.apache.org/native-doc/>
 - * <https://svn.apache.org/repos/asf/tomcat/native/>

This product contains the Maven wrapper scripts from 'Maven Wrapper', that provides an easy way to ensure a user has everything necessary to run the Maven build.

- * LICENSE:
 - * [license/LICENSE.mvn-wrapper.txt](#) (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/takari/maven-wrapper>

This product contains small piece of code to support AIX, taken from netbsd.

- * LICENSE:
 - * [license/LICENSE.aix-netbsd.txt](#) (OpenSSL License)
- * HOMEPAGE:
 - * <https://ftp.netbsd.org/pub/NetBSD/NetBSD-current/src/crypto/external/bsd/openssl/dist>

This product contains code from boringssl.

- * LICENSE (Combination ISC and OpenSSL license)
 - * [license/LICENSE.boringssl.txt](#) (Combination ISC and OpenSSL license)
- * HOMEPAGE:
 - * <https://boringssl.googlesource.com/boringssl/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.257 jackson-annotations 2.13.2

1.257.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.258 junit-jupiter-junit-jupiter-engine 5.7.2

1.258.1 Available under license :

```
import java.io.File
import java.net.URI
```

```
data class License(val name: String, val url: URI, val headerFile: File)
    Apache License
```

```
=====
```

```
_Version 2.0, January 2004_
_&lt;<https://www.apache.org/licenses/>&gt;_
```

```
### Terms and Conditions for use, reproduction, and distribution
```

```
#### 1. Definitions
```

License shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.

Licensors shall mean the copyright owner or entity authorized by the copyright

owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work,

provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets `[]` replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same printed page as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC
LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM
CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

* **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and

* **b)** in the case of each subsequent Contributor:

* **i)** changes to the Program, and

* **j)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations,

interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

* ****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

* **b)*** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

* **i)*** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

* **ii)*** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

* **iii)*** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

* **iv)*** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

* **a)*** it must be made available under this Agreement, or if the Program **(i)** is combined with other material in a separate file or files made available under a Secondary License, and **(ii)** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

* **b)*** a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: **a)** promptly notify the Commercial Contributor in writing of such claim, and **b)** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Open Source Licenses

=====

This product may include a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

[[contributors]]

== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.259 jetbrains-annotations 13.0

1.259.1 Available under license :

Copyright 2000-2012 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

1.260 scala 2.12.8

1.260.1 Available under license :

Scala

Copyright (c) 2002-2018 EPFL

Copyright (c) 2011-2018 Lightbend, Inc.

Scala includes software developed at
LAMP/EPFL (<https://lamp.epfl.ch/>) and
Lightbend, Inc. (<https://www.lightbend.com/>).

Licensed under the Apache License, Version 2.0 (the "License").
Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This software includes projects with other licenses -- see `doc/LICENSE.md`.
Scala includes the JLine library:

Copyright (c) 2002-2006, Marc Prud'hommeaux <mwp1@cornell.edu>
All rights reserved.

Redistribution and use in source and binary forms, with or
without modification, are permitted provided that the following
conditions are met:

Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer
in the documentation and/or other materials provided with
the distribution.

Neither the name of JLine nor the names of its contributors
may be used to endorse or promote products derived from this
software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING,
BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY
AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE

FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
{% if site.thisScalaVersion != site.latestScalaVersion %}
```

```
<div class="version-notice">This is the specification of a previous version of Scala. See the <a href="{ { site.baseurl } }/{ { site.latestScalaVersion } }"/>Scala { { site.latestScalaVersion } } spec</a></div>
```

```
{% endif %}
```

```
class C {
```

```
  val x: ((Int, Int) => Int) = (((a, b)) => a)
```

```
  val y: ((Int, Int, Int) => Int) = (((a, !!)) => a)
```

```
  val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)
```

```
}
```

Scala includes the ASM library.

Copyright (c) 2000-2011 INRIA, France Telecom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

not-a-legal-formal-parameter-tuple.scala:2: error: not a legal formal parameter.

Note: Tuples cannot be directly deconstructed in method or function parameters.

Either create a single parameter accepting the Tuple2,

or consider a pattern matching anonymous function: `{ case (a, b) => ... }

```
val x: ((Int, Int) => Int) = (((a, b)) => a)
```

^

not-a-legal-formal-parameter-tuple.scala:3: error: not a legal formal parameter.

Note: Tuples cannot be directly deconstructed in method or function parameters.

Either create a single parameter accepting the Tuple2,

or consider a pattern matching anonymous function: `{ case (param1, param2) => ... }

```
val y: ((Int, Int, Int) => Int) = (((a, !)) => a)
```

^

not-a-legal-formal-parameter-tuple.scala:4: error: not a legal formal parameter.

Note: Tuples cannot be directly deconstructed in method or function parameters.

Either create a single parameter accepting the Tuple3,

or consider a pattern matching anonymous function: `{ case (param1, ..., param3) => ... }

```
val z: ((Int, Int, Int) => Int) = (((a, NotAPatternVariableName, c)) => a)
```

^

three errors found

Scala is licensed under the [Apache License Version 2.0](<https://www.apache.org/licenses/LICENSE-2.0>).

Scala License

Copyright (c) 2002-2018 EPFL

Copyright (c) 2011-2018 Lightbend, Inc.

All rights reserved.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Other Licenses

This software includes projects with the following licenses, which are also included in the `licenses/` directory:

[Apache License](<http://www.apache.org/licenses/LICENSE-2.0.html>)

This license is used by the following third-party libraries:

* jansi

[BSD License](http://www.opensource.org/licenses/bsd-license.php)

This license is used by the following third-party libraries:

* jline

[BSD 3-Clause License](http://opensource.org/licenses/BSD-3-Clause)

This license is used by the following third-party libraries:

* asm

[MIT License](http://www.opensource.org/licenses/MIT)

This license is used by the following third-party libraries:

* jquery

* tools tooltip

Scala includes the Tools Tooltip library:

Copyright (c) 2009 Tero Piirainen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

(c) 2012-2014 GitHub

When using the GitHub logos, be sure to follow the GitHub logo guidelines (<https://github.com/logos>)

Font License: SIL OFL 1.1 (<http://scripts.sil.org/OFL>)

Applies to all font files

Code License: MIT (<http://choosealicense.com/licenses/mit/>)

Applies to all other files

Copyright (c) 2006, Ivan Sagalaev

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of highlight.js nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS AND CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Scala includes the JLine library, which includes the Jansi library.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical

transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable

by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use,

reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Scala includes the jQuery library:

Copyright (c) 2010 John Resig

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to

communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of

the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2002-2018 EPFL
Copyright (c) 2011-2018 Lightbend, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

(The MIT License)

Copyright (c) 2013 Greg Allen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the 'Software'), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED 'AS IS', WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.261 jersey-core-server 2.32

1.261.1 Available under license :

Notice for Jersey Core Server module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is

available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

org.objectweb.asm Version 8.0

* License: Modified BSD (<http://asm.objectweb.org/license.html>)

* Copyright: (c) 2000-2011 INRIA, France Telecom. All rights reserved.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Coyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>
- * Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available

in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF

TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of

this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these

conditions, and telling the user how to view a copy of this License.
(Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code

means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who

receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the

two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of

MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.262 ktor-bom 1.6.1

1.262.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.263 micronaut-groovy 3.1.0

1.263.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright 2017-2020 original authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* https://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

- * /opt/cola/permits/1331474157_1653512453.5913334/0/micronaut-runtime-groovy-3-1-0-sources-jar/io/micronaut/context/env/groovy/SetPropertyTransformer.groovy
- * /opt/cola/permits/1331474157_1653512453.5913334/0/micronaut-runtime-groovy-3-1-0-sources-jar/io/micronaut/context/env/groovy/ConfigurationEvaluator.groovy
- * /opt/cola/permits/1331474157_1653512453.5913334/0/micronaut-runtime-groovy-3-1-0-sources-jar/io/micronaut/context/env/groovy/GroovyPropertySourceLoader.java
- * /opt/cola/permits/1331474157_1653512453.5913334/0/micronaut-runtime-groovy-3-1-0-sources-jar/io/micronaut/context/env/groovy/package-info.java

1.264 io-swagger-swagger-annotations 2.1.12

1.264.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Bundle-Description: swagger-annotations

Automatic-Module-Name: io.swagger.v3.oas.annotations

Bundle-License: "Apache License 2.0";link="http://www.apache.org/licenses/LICENSE-2.0.html"

package: io.swagger

Bundle-SymbolicName: io.swagger.core.v3.swagger-annotations

implementation-version: 2.1.12

Built-By: runner

Bundle-ManifestVersion: 2

Bundle-DocURL: <https://github.com/swagger-api/swagger-core/modules/swagger-annotations>

Import-Package: io.swagger.v3.oas.annotations,io.swagger.v3.oas.annotations.enums,io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.headers,io.swagger.v3.oas.annotations.info,io.swagger.v3.oas.annotations.links,io.swagger.v3.oas.annotations.media,io.swagger.v3.oas.annotations.parameters,io.swagger.v3.oas.annotations.responses,io.swagger.v3.oas.annotations.security,io.swagger.v3.oas.annotations.servers,io.swagger.v3.oas.annotations.tags

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"

Bundle-Developers: fehguy;email="fehguy@gmail.com";name="Tony Tam",webbron;email="webbron@gmail.com";name="Ron Ratovsky"

url: <https://github.com/swagger-api/swagger-core/modules/swagger-annotations>

mode: development

Export-Package: io.swagger.v3.oas.annotations;uses:="io.swagger.v3.oas.annotations.enums,io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.info,io.swagger.v3.oas.annotations.media,io.swagger.v3.oas.annotations.parameters,io.swagger.v3.oas.annotations.responses,io.swagger.v3.oas.annotations.security,io.swagger.v3.oas.annotations.servers,io.swagger.v3.oas.annotations.tags";version="2.1.12",io.swagger.v3.oas.annotations.callbacks;uses:="io.swagger.v3.oas.annotat

ions,io.swagger.v3.oas.annotations.extensions";version="2.1.12",io.swagger.v3.oas.annotations.enums;version="2.1.12",io.swagger.v3.oas.annotations.extensions;version="2.1.12",io.swagger.v3.oas.annotations.headers;uses:="io.swagger.v3.oas.annotations.media";version="2.1.12",io.swagger.v3.oas.annotations.info;uses:="io.swagger.v3.oas.annotations.extensions";version="2.1.12",io.swagger.v3.oas.annotations.links;uses:="io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.servers";version="2.1.12",io.swagger.v3.oas.annotations.media;uses:="io.swagger.v3.oas.annotations,io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.headers";version="2.1.12",io.swagger.v3.oas.annotations.parameters;uses:="io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.media";version="2.1.12",io.swagger.v3.oas.annotations.responses;uses:="io.swagger.v3.oas.annotations.extensions,io.swagger.v3.oas.annotations.headers,io.swagger.v3.oas.annotations.links,io.swagger.v3.oas.annotations.media";version="2.1.12",io.swagger.v3.oas.annotations.security;uses:="io.swagger.v3.oas.annotations.enums,io.swagger.v3.oas.annotations.extensions";version="2.1.12",io.swagger.v3.oas.annotations.servers;uses:="io.swagger.v3.oas.annotations.extensions";version="2.1.12",io.swagger.v3.oas.annotations.tags;uses:="io.swagger.v3.oas.annotations,io.swagger.v3.oas.annotations.extensions";version="2.1.12"

Bundle-Name: swagger-annotations

Bundle-Version: 2.1.12

Bundle-SCM: url="https://github.com/swagger-api/swagger-core/modules/swagger-annotations",connection="scm:git:git@github.com:swagger-api/swagger-core.git/modules/swagger-annotations",developer-connection="scm:git:git@github.com:swagger-api/swagger-core.git/modules/swagger-annotations",tag=HEAD

Created-By: Apache Maven 3.8.4

Build-Jdk: 1.8.0_312

Found in path(s):

* /opt/cola/permits/1340031602_1654736977.0737062/0/swagger-annotations-2-1-12-jar/META-INF/MANIFEST.MF

1.265 kafka-schema-registry-client 6.2.0

1.265.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://maven.apache.org/POM/4.0.0"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>
```

```

    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-parent</artifactId>
    <version>6.2.0</version>
  </parent>

  <licenses>
    <license>
      <name>Apache License 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
      <distribution>repo</distribution>
    </license>
  </licenses>

  <artifactId>kafka-schema-registry-client</artifactId>
  <packaging>jar</packaging>
  <name>kafka-schema-registry-client</name>

  <dependencies>
    <dependency>
      <groupId>org.apache.kafka</groupId>
      <artifactId>kafka-clients</artifactId>
    </dependency>
    <dependency>
      <groupId>org.apache.avro</groupId>
      <artifactId>avro</artifactId>
    </dependency>
    <dependency>
      <groupId>com.fasterxml.jackson.core</groupId>
      <artifactId>jackson-databind</artifactId>
    </dependency>
    <dependency>
      <groupId>jakarta.ws.rs</groupId>
      <artifactId>jakarta.ws.rs-api</artifactId>
    </dependency>
    <dependency>
      <groupId>org.glassfish.jersey.core</groupId>
      <artifactId>jersey-common</artifactId>
      <version>${jersey.version}</version>
    </dependency>
    <dependency>
      <groupId>org.easymock</groupId>
      <artifactId>easymock</artifactId>
      <scope>test</scope>
    </dependency>
    <dependency>
      <groupId>org.powermock</groupId>
      <artifactId>powermock-module-junit4</artifactId>
      <scope>test</scope>

```

```

</dependency>
<dependency>
  <groupId>org.powermock</groupId>
  <artifactId>powermock-api-easymock</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>io.swagger</groupId>
  <artifactId>swagger-annotations</artifactId>
</dependency>
<dependency>
  <groupId>com.google.guava</groupId>
  <artifactId>guava</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>
</project>

```

Found in path(s):

```

* /opt/cola/permits/1340188481_1654734945.6160817/0/kafka-schema-registry-client-6-2-0-jar/META-
INF/maven/io.confluent/kafka-schema-registry-client/pom.xml

```

1.266 asm 8.0.1

1.266.1 Available under license :

No license file was found, but licenses were detected in source scan.

```

// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.

```

Found in path(s):

```

* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-
jar/org/objectweb/asm/CurrentFrame.java

```

* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Label.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Type.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/FieldVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ByteVector.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/RecordComponentVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/AnnotationVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Handler.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Constants.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/signature/SignatureReader.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/SymbolTable.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/signature/SignatureWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/AnnotationWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Handle.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Attribute.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/FieldWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/TypePath.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/MethodVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/signature/SignatureVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/MethodWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ClassVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Edge.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/MethodTooLargeException.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ConstantDynamic.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ClassTooLargeException.java

* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/RecordComponentWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Opcodes.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ClassReader.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ModuleWriter.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Context.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/TypeReference.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ModuleVisitor.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Symbol.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/Frame.java
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/ClassWriter.java

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

- * 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
- * 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/signature/package.html
* /opt/ws_local/PERMITS_SQL/1093933321_1601047939.19/0/asm-8-0-1-sources-1-jar/org/objectweb/asm/package.html

1.267 dropwizard-jackson-support 2.0.18

1.267.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation,

and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s)

with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.
Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.268 jackson-datatype-guava 2.13.2

1.268.1 Available under license :

This copy of Jackson JSON processor `jackson-datatype-guava` module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.269 google-gson 2.8.6

1.269.1 Available under license :

Google Gson

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2008-2011 Google Inc.

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.270 googleapis-core 0.0.3

1.270.1 Available under license :

Copyright 2020 Google LLC

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.271 pact-jvm-core-model 4.0.10

1.271.1 Available under license :

Apache-2.0

1.272 guava 31.0.1-jre

1.272.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/**

* Holder for web specializations of methods of { @code Floats }. Intended to be empty for regular
* version.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/FloatsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2007 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/*

* This following method is a modified version of one found in
* <http://gee.cs.oswego.edu/cgi-bin/viewcvs.cgi/jsr166/src/test/tck/AbstractExecutorServiceTest.java?revision=1.30>
* which contained the following notice:
*
* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to
* the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>
*
* Other contributors include Andrew Wright, Jeffrey Hayes, Pat Fisher, Mike Judd.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/MoreExecutors.java

No license file was found, but licenses were detected in source scan.

/*

* Written by Doug Lea with assistance from members of JCP JSR-166
* Expert Group and released to the public domain, as explained at
* <http://creativecommons.org/publicdomain/zero/1.0/>
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/LongAdder.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AtomicDoubleArray.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/Striped64.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/Striped64.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/LongAdder.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/**

* Holder for web specializations of methods of { @code Doubles }. Intended to be empty for regular
* version.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/DoublesMethodsForWeb.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AtomicLongMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/GwtTransient.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2013 The Guava Authors
*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MultimapBuilder.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapEntry.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MoreFiles.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2014 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
* express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/SubscriberRegistry.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/MoreObjects.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Subscriber.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenerCallQueue.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Dispatcher.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/TrustedListenableFutureTask.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/Quantiles.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2007 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Functions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Objects.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableSoftReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/AllowConcurrentEvents.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/DeadEvent.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ExecutionList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MultiInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LittleEndianDataInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenableFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Charsets.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/EventBus.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Defaults.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Predicates.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Closeables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/Subscribe.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CharStreams.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/AsyncEventBus.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Supplier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LineBuffer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Preconditions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Resources.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LineReader.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/LittleEndianDataOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Throwables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Predicate.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/ByteStreams.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableReferenceQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Files.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableWeakReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/AbstractIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Primitives.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizableReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CountingOutputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FinalizablePhantomReference.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EnumMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/CountingInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Flushables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/DirectExecutor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Interners.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Suppliers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Function.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2012 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ByteSource.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/AbstractByteHasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/AbstractInvocationHandler.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/Parameter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ServiceManager.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/StandardSystemProperty.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/StatsAccumulator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CartesianList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/LongAddable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/TypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/html/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/hash/ChecksumHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ByteSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/TypeCapture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/LinearTransformation.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredKeyMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/ClassPath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/LongAddables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/package-
info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/BaseEncoding.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/SmoothRateLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/SipHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/MutableTypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/CharSource.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/Stats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableRangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/PairedStatsAccumulator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/FileWriteMode.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableRangeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/CharSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/LongAddable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/LongAddables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/RateLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/math/PairedStats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/Invokable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/ImmutableTypeToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/Closer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ListenableScheduledFuture.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2010 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/package-info.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/annotations/Beta.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ListeningExecutorService.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ThreadFactoryBuilder.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Ascii.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Equivalence.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Strings.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/UncaughtExceptionHandler.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Atomics.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ContiguousSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SortedLists.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingBlockingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Monitor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/annotations/package-info.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2007 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Multiset.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Iterables.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedSetMultimap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MapDifference.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Sets.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashMultiset.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeMultimap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractIterator.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/package-info.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MutableClassToInstanceMap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/SetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Multisets.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/BiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSortedMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ReverseOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractMapEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ByFunctionOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Ordering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingMapEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/EnumHashBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Interner.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingObject.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Multimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/LinkedListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ExplicitOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ComparatorOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractMapBasedMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Lists.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/RegularImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/EnumBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/LinkedHashMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Iterators.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/NaturalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Multimaps.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/NullsFirstOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/NullsLastOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/UsingToStringOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Maps.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ClassToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ConcurrentHashMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ArrayListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TreeMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingConcurrentMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractMapBasedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/ForwardingCollection.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SingletonImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractSortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/HashMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/LexicographicalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CompoundOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Synchronized.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ReverseNaturalOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/LinkedHashMultiset.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/ImmutableIntArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ForwardingCondition.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ForwardingLock.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/ImmutableDoubleArray.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/hash/AbstractHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/ImmutableLongArray.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2016 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/collect/LinkedHashMultimapGwtSerializationDependencies.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/DirectedNetworkConnections.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/ElementOrder.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/DirectedMultiNetworkConnections.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/ImmutableValueGraph.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/collect/CollectCollectors.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/MutableValueGraph.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/AbstractGraphBuilder.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/StandardMutableGraph.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/collect/ImmutableMultisetGwtSerializationDependencies.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/collect/ArrayListMultimapGwtSerializationDependencies.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/GraphBuilder.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/StandardMutableValueGraph.java
```

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Comparators.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EndpointPairIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ValueGraphBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MapRetrievalCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ForwardingGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/MoreCollectors.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedMultiNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/GraphConstants.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ForwardingValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardMutableNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RangeGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/StandardValueGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MapIteratorCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractUndirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/NetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashMultimapGwtSerializationDependencies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/MultiEdgesConnecting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/NetworkBuilder.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ForwardingNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EdgesConnecting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/AbstractDirectedNetworkConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/GraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/DirectedGraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/UndirectedGraphConnections.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/EndpointPair.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Internal.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Platform.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2007 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

/**

* Returns an array containing all of the elements in the specified collection. This method
* returns the elements in the order they are returned by the collection's iterator. The returned
* array is "safe" in that no references to it are maintained by the collection. The caller is
* thus free to modify the returned array.

*

* <p>This method assumes that the collection size doesn't change while the method is running.

*

* <p>TODO(kevinb): support concurrently modified collections?

*

* @param c the collection for which to return an array of elements

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ObjectArrays.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

/*

* This method was rewritten in Java from an intermediate step of the Murmur hash function in
* <http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp>, which contained the
* following header:

*

* MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
* hereby disclaims copyright to this source code.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/base/SmallCharMatcher.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you

* may not use this file except in compliance with the License. You may

* obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or

* implied. See the License for the specific language governing

* permissions and limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/Streams.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2012 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/CompactHashSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/FilteredMultimap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/DescendingImmutableSortedSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/AbstractNavigableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredKeySetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CompactLinkedHashSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/UnmodifiableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/EvictingQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CompactHashMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TreeTraverser.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/DescendingMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableEnumMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredEntryMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableAsList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingNavigableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TransformedListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredKeyListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TreeRangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CompactLinkedHashMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingBlockingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredEntrySetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RangeMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/AbstractSortedKeySortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AllEqualOrdering.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingNavigableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ForwardingBlockingDeque.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TransformedIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SortedMultisetBridge.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingImmutableMap.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2013 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
* express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/TypeVisitor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/eventbus/SubscriberExceptionHandler.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/VerifyException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Verify.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/thirdparty/publicsuffix/PublicSuffixType.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/FilteredMultimapValues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/eventbus/SubscriberExceptionContext.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/io/CharSequenceReader.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/WrappingScheduledExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/HashingInputStream.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Utf8.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Runnables.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2011 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,
either
* express or implied. See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/GeneralRange.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedMultisetFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractRangeSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SortedIterable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Count.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SortedIterables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RangeSet.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Platform.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ReaderInputStream.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/InterruptibleTask.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AsyncCallable.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/LittleEndianByteArray.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ConsumingQueueIterator.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/MacHashFunction.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/CombinedFuture.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AggregateFutureState.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/FarmHashFingerprint64.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/ImmutableSupplier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/JdkBackedImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ExecutionSequencer.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/JdkBackedImmutableSet.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/AbstractStreamingHasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/Cache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/AbstractLoadingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/CacheBuilderSpec.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/BloomFilter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/UnsignedLong.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/cache/ForwardingLoadingCache.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/DescendingImmutableSortedMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/HostAndPort.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/PrimitiveSink.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/IntMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Hasher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/PairwiseEquivalence.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/LoadingCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ParseRequest.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Hashing.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/DoubleUtils.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/Weigher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedInteger.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractScheduledService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AbstractListeningExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedInts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ExecutionError.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Funnel.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/TypeParameter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/BloomFilterStrategies.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/CycleDetectingLockFactory.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/MediaType.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/Uninterruptibles.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/HttpHeaders.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalCause.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/package-info.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/MathPreconditions.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Ticker.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Absent.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingExecutorService.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/EmptyContiguousSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/DoubleMath.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalListener.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/AbstractSortedMultiset.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Enums.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractCompositeHashFunction.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Present.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Funnels.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/UnsignedLongs.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashingOutputStream.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Optional.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/BigIntegerMath.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractHasher.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/BoundType.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/TreeRangeSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalListeners.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ForwardingCache.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/FutureCallback.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashCode.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Murmur3_128HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/LongMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/AbstractNonStreamingHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/MessageDigestHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Murmur3_32HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Queues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/AbstractCache.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ForwardingListeningExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/Crc32cHashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/WrappingExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/UncheckedExecutionException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/HashFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheStats.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/RemovalNotification.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/AsyncFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/RegularContiguousSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListeningScheduledExecutorService.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/FunctionalEquivalence.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/Types.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/CacheLoader.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors.

*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/package-info.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2014 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TopKSelector.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/Graphs.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/PredecessorsFunction.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/ImmutableNetwork.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/Graph.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/MutableGraph.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/graph/Network.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/MutableNetwork.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/SuccessorsFunction.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/InsecureRecursiveDeleteException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/ImmutableGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/RecursiveDeleteOption.java
No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2009 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/**
 * Not supported. <b>You are attempting to create a map that may contain a non-{@code Comparable}
 * key.</b> Proper calls will resolve to the version in {@code ImmutableSortedMap}, not this dummy
 * version.
 *
 * @throws UnsupportedOperationException always
 * @deprecated <b>Pass a key of type {@code Comparable} to use {@link
 *   ImmutableSortedMap#of(Comparable, Object)}.</b>
 */
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedMapFauxverideShim.java
No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2016 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
```

```
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/JdkPattern.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/CommonPattern.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/CommonMatcher.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/PatternCompiler.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2020 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for web specializations of methods of { @code Ints }. Intended to be empty for regular
* version.
*/
```

Found in path(s):

```
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/IntsMethodsForWeb.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright (C) 2006 The Guava Authors
*
```

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/FuturesGetChecked.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/UncheckedTimeoutException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/TimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/FluentFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ImmediateFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/CharEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/TypeToken.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/PatternFilenameFilter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/annotations/VisibleForTesting.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/SimpleTimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/FakeTimeLimiter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/AppendableWriter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Futures.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AbstractCatchingFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/TimeoutFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/GwtFuturesCatchingSpecialization.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/CharEscaperBuilder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/AbstractTransformFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/CollectionFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/CaseFormat.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/AggregateFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/GwtFluentFutureCatchingSpecialization.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2009 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the  
* License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND,  
either  
* express or implied. See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedAsList.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2005 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except  
* in compliance with the License. You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software distributed under the License  
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either  
express  
* or implied. See the License for the specific language governing permissions and limitations under  
* the License.  
*/
```

Found in path(s):
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/reflect/Reflection.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2009 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/ForwardingListenableFuture.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/Service.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/cache/CacheBuilder.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/ForwardingFuture.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/MapMakerInternalMap.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/annotations/GwtCompatible.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/annotations/GwtIncompatible.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/cache/LocalCache.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/AbstractExecutionThreadService.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/AbstractIdleService.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/cache/ReferenceEntry.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/net/InternetDomainName.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/html/HtmlEscapers.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/AbstractService.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/Cut.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ForwardingFluentFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/Escapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/SignedBytes.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/primitives/UnsignedBytes.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/net/HostSpecifier.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/ArrayBasedCharEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Splitter.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/MapMaker.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/DenseImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/net/UrlEscapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ByteArrayDataOutput.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Callables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/ArrayBasedEscaperMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/reflect/TypeResolver.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ByteArrayDataInput.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/escape/ArrayBasedUnicodeEscaper.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SparseImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/xml/XmlEscapers.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/LineProcessor.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/JdkFutureAdapters.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/SettableFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/ByteProcessor.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2015 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/package-info.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableBiMapFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/CollectSpliterators.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2018 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/JdkBackedImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/IndexedImmutableSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/BaseImmutableMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/JdkBackedImmutableMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2010 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/MinMaxPriorityQueue.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingImmutableCollection.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractSequentialIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/UnmodifiableListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingSortedSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RowSortedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SortedMapDifference.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2009 The Guava Authors

*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableClassToInstanceMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/DiscreteDomain.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedSetFauxverideShim.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ForwardingTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableEnumSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableAsList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/AbstractIndexedListIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/EmptyImmutableSetMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SingletonImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TableCollectors.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableList.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ComparisonChain.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/ArrayTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ComputationException.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SingletonImmutableList.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Copyright (C) 2021 The Guava Authors  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*/
```

Found in path(s):

```
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/escape/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/io/ElementTypesAreNonnullByDefault.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/base/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/net/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/xml/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/util/concurrent/ElementTypesAreNonnullByDefault.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/io/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/escape/ElementTypesAreNonnullByDefault.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/eventbus/ElementTypesAreNonnullByDefault.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/reflect/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/primitives/ParametricNullness.java  
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-  
jar/com/google/common/graph/ElementTypesAreNonnullByDefault.java
```

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/xml/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/reflect/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/eventbus/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/math/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/cache/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/ParametricNullness.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/hash/ElementTypesAreNonnullByDefault.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/html/ParametricNullness.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either

express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

/**

* Holder for web specializations of methods of { @code Shorts }. Intended to be empty for regular

* version.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/primitives/ShortsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either

express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

/*

* This method was written by Doug Lea with assistance from members of JCP JSR-166 Expert Group

* and released to the public domain, as explained at

* <http://creativecommons.org/licenses/publicdomain>

*

* As of 2010/06/11, this method is identical to the (package private) hash method in OpenJDK 7's

* `java.util.HashMap` class.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/util/concurrent/Striped.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2017 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/AbstractBaseGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ClosingFuture.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/BaseGraph.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/graph/Traverser.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2016 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/
/**
* Holder for extra methods of {@code Objects} only in web. Intended to be empty for regular
* version.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/ExtraObjectsMethodsForWeb.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2020 The Guava Authors
*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software distributed under the License
* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express
* or implied. See the License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/BigDecimalMath.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/OverflowAvoidingLockSupport.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/hash/Java8Compatibility.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/math/ToDoubleRounder.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/Java8Compatibility.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/io/Java8Compatibility.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2009 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

/**

* Outer class that exists solely to let us write {@code Partially.GwtIncompatible} instead of plain

* {@code GwtIncompatible}. This is more accurate for {@link Futures#catching}, which is available

* under GWT but with a slightly different signature.

*

* <p>We can't use {@code PartiallyGwtIncompatible} because then the GWT compiler wouldn't recognize

* it as a {@code GwtIncompatible} annotation. And for {@code Futures.catching}, we need the GWT

* compiler to autostrip the normal server method in order to expose the special, inherited GWT
* version.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/Partially.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2020 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/util/concurrent/ServiceManagerBridge.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2021 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/base/NullnessCasts.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/NullnessCasts.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/NullnessCasts.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 The Guava Authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Platform.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapEntrySet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapValues.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Tables.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/Collections2.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMultiset.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/HashBasedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CollectPreconditions.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/thirdparty/publicsuffix/PublicSuffixPatterns.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/ImmutableMapKeySet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/collect/StandardRowSortedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/PeekingIterator.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Table.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableEntry.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/RegularImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/SingletonImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Serialization.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableSortedSet.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableBiMap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/StandardTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/EmptyImmutableListMultimap.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/ImmutableCollection.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Range.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/TreeBasedTable.java
* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/UnmodifiableIterator.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2008 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except

* in compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License

* is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
express

* or implied. See the License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/internal/Finalizer.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Booleans.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/thirdparty/publicsuffix/TrieParser.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Ints.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/ListenableFutureTask.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Chars.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Shorts.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/Escaper.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Floats.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Stopwatch.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/FileBackedOutputStream.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/CharMatcher.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/escape/UnicodeEscaper.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/FluentIterable.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/io/MultiReader.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Converter.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Doubles.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/net/InetAddresses.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Longs.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/primitives/Bytes.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/base/Joiner.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/util/concurrent/SequentialExecutor.java
- * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-

jar/com/google/common/net/PercentEscaper.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2011 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not

* use this file except in compliance with the License. You may obtain a copy of

* the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

* License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMultiset.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/SortedMultisets.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/graph/IncidentEdgeSet.java

* /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-jar/com/google/common/collect/CompactHashing.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2008 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
/*
 * This method was rewritten in Java from an intermediate step of the Murmur hash function in
 * http://code.google.com/p/smhasher/source/browse/trunk/MurmurHash3.cpp, which contained the
 * following header:
 *
 * MurmurHash3 was written by Austin Appleby, and is placed in the public domain. The author
 * hereby disclaims copyright to this source code.
 */
```

Found in path(s):

```
*/
 * /opt/cola/permits/1288520115_1647351767.93/0/guava-31-0-1-jre-sources-
jar/com/google/common/collect/Hashing.java
```

1.273 asm-tree 8.0.1

1.273.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
```


Found in path(s):

- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/MethodInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LocalVariableNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/TypeInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/Util.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/FieldNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/TypeAnnotationNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/MethodNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ModuleProvideNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/InvokeDynamicInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LookupSwitchInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ClassNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LdcInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ModuleNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/IncnInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/VarInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/FrameNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/TableSwitchInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/JumpInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/FieldInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/MultiANewArrayInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ModuleOpenNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/IntInsnNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LineNumberNode.java
- * /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/InsnNode.java

* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/AnnotationNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LabelNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/LocalVariableAnnotationNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/AbstractInsnNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/InnerClassNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ModuleRequireNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/InsnList.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ModuleExportNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/TryCatchBlockNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/ParameterNode.java
* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-jar/org/objectweb/asm/tree/RecordComponentNode.java

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

- * 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
- * 3. Neither the name of the copyright holders nor the names of its
* contributors may be used to endorse or promote products derived from
* this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS"
* AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
* IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
* ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE
* LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR
* CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF
* SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS
* INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
* CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

* ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF
* THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1104664895_1604644145.26/0/asm-tree-8-0-1-sources-1-
jar/org/objectweb/asm/tree/package.html

1.274 micronaut-rxjava-2 1.2.1

1.274.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work

(an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses

granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]"

replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.275 jdbi3-bom 3.20.1

1.275.1 Available under license :

Jdbi 1 & 2

- Brian McCallister
- Martin Traverso
- Henning Schmiedehausen
- Jax Law
- maniax, for finding a nasty bug in named parameter substitution
- Robert Sfeir, for batch processing feature requests
- Henry Yandell, for `in` clause expansion feature request
- Patrick Burlinson, for showing that `DBIException` should extend `RuntimeException`
- Thomas Risberg, for the Spring `DataAccessException` wrapping code
- Simone Gianni, for finding and solving a nasty `setNull` bug
- Thomas Dudziak

Jdbi 3

- Steven Schlansker
- Matthew Hall
- Artem Prigoda
- Marnick L'Eau

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

JDBI

Copyright (C) 2004 - 2015 Brian McCallister

1.276 netty-transport-native-unix-common

4.1.74.Final

1.276.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2016 The Netty Project

~

~ The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

~

~ <https://www.apache.org/licenses/LICENSE-2.0>

~

~ Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

-->

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/META-INF/maven/io.netty/netty-transport-native-unix-common/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2018 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License, version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/Buffer.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_buffer.c
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/PreferredDirectByteBufferAllocator.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_buffer.h

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/package-info.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/Unix.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/IovArray.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/UnixChannelOption.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2020 The Netty Project

*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix.c

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_jni.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/UnixChannelUtil.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2016 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,

* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_limits.c
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/LimitsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_limits.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_util.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/Limits.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/PeerCredentials.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/ErrorsStaticallyReferencedJniMethods.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/SocketWritableByteChannel.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_util.c

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2021 The Netty Project
*
* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.
*/

Found in path(s):

- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainDatagramSocketAddress.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainDatagramChannelConfig.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/SegmentedDatagramPacket.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainDatagramPacket.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainDatagramChannel.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/NativeInetAddress.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_filedescriptor.c
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainSocketAddress.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_errors.c
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_socket.c
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DatagramSocketAddress.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/ServerDomainSocketChannel.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/UnixChannel.java
- * /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-

2-jar/netty_unix_filedescriptor.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainSocketChannel.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainSocketChannelConfig.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_socket.h
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/FileDescriptor.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/DomainSocketReadMode.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/Errors.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/io/netty/channel/unix/Socket.java
* /opt/cola/permits/1340188451_1654650562.854448/0/netty-transport-native-unix-common-4-1-74-final-sources-2-jar/netty_unix_errors.h

1.277 testcontainers-bom 1.16.2

1.277.1 Available under license :

ibmcom/db2:11.5.0.0a
The MIT License (MIT)

Copyright (c) 2017 - 2019 G DATA Software AG and other authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2017 Capital One Services, LLC and other authors

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License (MIT)

Copyright (c) 2015-2019 Richard North

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

neo4j:3.5.0-enterprise

a

b

mcr.microsoft.com/mssql/server:2017-CU12

1.278 fabric8-::-kubernetes-model-::-batch

4.10.3

1.278.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.279 guava-listenablefuture-only 9999.0-empty-to-avoid-conflict-with-guava

1.279.1 Available under license :

Found license 'GNU Lesser General Public License' in '// This library is free software; you can redistribute it and/or // modify it under the terms of the GNU Lesser General Public // License as published by the Free Software Foundation; either // version 2.1 of the License, or (at your option) any later version. // This library is distributed in the hope that it will be useful, // but WITHOUT ANY WARRANTY; without even the implied warranty of // MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU // Lesser General Public License for more details. // You should have received a copy of the GNU Lesser General Public * This grammar is in the PUBLIC DOMAIN'

1.280 fabric8-::-kubernetes-model-::-coordination 4.10.3

1.280.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.281 kotlinox-coroutines-io 0.1.16

1.281.1 Available under license :

Apache License

Version 2.0, January 2004

<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.282 cloudevents---kafka-transport-binding

2.1.1

1.282.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2018-Present The CloudEvents Authors
~ <p>
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~ <p>
~ http://www.apache.org/licenses/LICENSE-2.0
~ <p>
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
~
-->
```

Found in path(s):

* /opt/cola/permits/1337069740_1654221784.6351657/0/cloudevents-kafka-2-1-1-jar/META-INF/maven/io.cloudevents/cloudevents-kafka/pom.xml

1.283 micronaut-grpc 3.1.3

1.283.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.284 asm-analysis 8.0.1

1.284.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions  
// are met:  
// 1. Redistributions of source code must retain the above copyright  
// notice, this list of conditions and the following disclaimer.  
// 2. Redistributions in binary form must reproduce the above copyright  
// notice, this list of conditions and the following disclaimer in the  
// documentation and/or other materials provided with the distribution.  
// 3. Neither the name of the copyright holders nor the names of its  
// this software without specific prior written permission.
```

Found in path(s):

```
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/SourceValue.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/Subroutine.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/BasicVerifier.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/Frame.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/BasicValue.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/AnalyzerException.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/Interpreter.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/SmallSet.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/Analyzer.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/SimpleVerifier.java  
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-  
jar/org/objectweb/asm/tree/analysis/Value.java
```

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.  
// Redistribution and use in source and binary forms, with or without  
// modification, are permitted provided that the following conditions
```

```
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new { @link BasicInterpreter } for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the { @link #BasicInterpreter(int) }
 * version.
 */
```

Found in path(s):

```
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-
jar/org/objectweb/asm/tree/analysis/BasicInterpreter.java
```

No license file was found, but licenses were detected in source scan.

```
// All rights reserved.
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. Neither the name of the copyright holders nor the names of its
// this software without specific prior written permission.
/**
 * Constructs a new { @link SourceInterpreter } for the latest ASM API version. <i>Subclasses must
 * not use this constructor</i>. Instead, they must use the { @link #SourceInterpreter(int) }
 * version.
 */
```

Found in path(s):

```
* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-
jar/org/objectweb/asm/tree/analysis/SourceInterpreter.java
```

No license file was found, but licenses were detected in source scan.

2011 INRIA, France Telecom

```
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
```

- * notice, this list of conditions and the following disclaimer.
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * 3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.
- *
- * THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Found in path(s):

* /opt/cola/permits/1104664774_1604644122.5/0/asm-analysis-8-0-1-sources-1-jar/org/objectweb/asm/tree/analysis/package.html

1.285 dropwizard-metrics-support 2.0.18

1.285.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.286 jersey-ext-metainf-services 2.32

1.286.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Coyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

* License: Apache License, 2.0

* Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement

or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights

in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software

or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE

file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute

such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1

and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new

versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it

free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.287 io-grpc-grpc-netty 1.37.0

1.287.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.288 arrow-core-extensions 0.9.0

1.288.1 Available under license :

Apache-2.0

1.289 pact-jvm-core-support 4.0.10

1.289.1 Available under license :

Apache-2.0

1.290 okhttp-logging-interceptor 3.12.12

1.290.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1294625109_1649800590.01/0/logging-interceptor-3-12-12-sources-jar/okhttp3/logging/HttpLoggingInterceptor.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

* /opt/cola/permits/1294625109_1649800590.01/0/logging-interceptor-3-12-12-sources-jar/okhttp3/logging/LoggingEventListener.java

1.291 metrics-integration-for-jersey-2.x 4.1.17

1.291.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.codahale.metrics

Build-Jdk-Spec: 1.8

Bundle-Description: Metrics is a Java library which gives you unparalleled insight into what your code does in production. Metrics provides a powerful toolkit of ways to measure the behavior of critical components in your production environment.

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>

Bundle-ManifestVersion: 2

Bundle-Name: Metrics Core

Bundle-SymbolicName: io.dropwizard.metrics.core

Bundle-Version: 4.1.17

Created-By: Apache Maven Bundle Plugin

Export-Package: com.codahale.metrics;uses:="org.slf4j";version="4.1.17"

"

Implementation-Title: Metrics Core

Implementation-Version: 4.1.17

Import-Package: org.slf4j;version="[1.6.0,2.0.0)"

Private-Package: com.codahale.metrics

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"

Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188703_1654734276.6414893/0/metrics-core-4-1-17-jar/META-INF/MANIFEST.MF

1.292 hystrix-core 1.5.18

1.292.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR

PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.293 kotlin-logging 1.6.26

1.293.1 Available under license :

Copyright (c) 2016-2018 Ohad Shai

This software is licensed under the Apache 2 license, quoted below.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made,

use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions

for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability

incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright (c) 2016-2018 Ohad Shai

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.294 findbugs-jsr305 3.0.2

1.294.1 Available under license :

License

THE WORK (AS DEFINED BELOW) IS PROVIDED UNDER THE TERMS OF THIS CREATIVE COMMONS PUBLIC LICENSE ("CCPL" OR "LICENSE"). THE WORK IS PROTECTED BY COPYRIGHT AND/OR OTHER APPLICABLE LAW. ANY USE OF THE WORK OTHER THAN AS AUTHORIZED UNDER THIS LICENSE OR COPYRIGHT LAW IS PROHIBITED.

BY EXERCISING ANY RIGHTS TO THE WORK PROVIDED HERE, YOU ACCEPT AND AGREE TO BE BOUND BY THE TERMS OF THIS LICENSE. THE LICENSOR GRANTS YOU THE RIGHTS CONTAINED HERE IN CONSIDERATION OF YOUR ACCEPTANCE OF SUCH TERMS AND CONDITIONS.

1. Definitions

1. "Collective Work" means a work, such as a periodical issue, anthology or encyclopedia, in which the Work in its

entirety in unmodified form, along with a number of other contributions, constituting separate and independent works in themselves, are assembled into a collective whole. A work that constitutes a Collective Work will not be considered a Derivative Work (as defined below) for the purposes of this License.

2. "Derivative Work" means a work based upon the Work or upon the Work and other pre-existing works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which the Work may be recast, transformed, or adapted, except that a work that constitutes a Collective Work will not be considered a Derivative Work for the purpose of this License. For the avoidance of doubt, where the Work is a musical composition or sound recording, the synchronization of the Work in timed-relation with a moving image ("synching") will be considered a Derivative Work for the purpose of this License.

3. "Licensor" means the individual or entity that offers the Work under the terms of this License.

4. "Original Author" means the individual or entity who created the Work.

5. "Work" means the copyrightable work of authorship offered under the terms of this License.

6. "You" means an individual or entity exercising rights under this License who has not previously violated the terms of this License with respect to the Work, or who has received express permission from the Licensor to exercise rights under this License despite a previous violation.

2. Fair Use Rights. Nothing in this license is intended to reduce, limit, or restrict any rights arising from fair use, first sale or other limitations on the exclusive rights of the copyright owner under copyright law or other applicable laws.

3. License Grant. Subject to the terms and conditions of this License, Licensor hereby grants You a worldwide, royalty-free, non-exclusive, perpetual (for the duration of the applicable copyright) license to exercise the rights in the Work as stated below:

1. to reproduce the Work, to incorporate the Work into one or more Collective Works, and to reproduce the Work as incorporated in the Collective Works;
2. to create and reproduce Derivative Works;
3. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission the Work including as incorporated in Collective Works;
4. to distribute copies or phonorecords of, display publicly, perform publicly, and perform publicly by means of a digital audio transmission Derivative Works.
- 5.

For the avoidance of doubt, where the work is a musical composition:

1. Performance Royalties Under Blanket Licenses. Licensor waives the exclusive right to collect, whether individually or via a performance rights society (e.g. ASCAP, BMI, SESAC), royalties for the public performance or public digital performance (e.g. webcast) of the Work.

2. Mechanical Rights and Statutory Royalties. Licensor waives the exclusive right to collect, whether individually or via a music rights agency or designated agent (e.g. Harry Fox Agency), royalties for any phonorecord You create from the Work ("cover version") and distribute, subject to the compulsory license created by 17 USC Section 115 of the US Copyright Act (or the equivalent in other jurisdictions).

6. Webcasting Rights and Statutory Royalties. For the avoidance of doubt, where the Work is a sound recording, Licensor waives the exclusive right to collect, whether individually or via a performance-rights society (e.g. SoundExchange), royalties for the public digital performance (e.g. webcast) of the Work, subject to the compulsory license created by 17 USC Section 114 of the US Copyright Act (or the equivalent in other jurisdictions).

The above rights may be exercised in all media and formats whether now known or hereafter devised. The above

rights include the right to make such modifications as are technically necessary to exercise the rights in other media and formats. All rights not expressly granted by Licensor are hereby reserved.

4. Restrictions. The license granted in Section 3 above is expressly made subject to and limited by the following restrictions:

1. You may distribute, publicly display, publicly perform, or publicly digitally perform the Work only under the terms of this License, and You must include a copy of, or the Uniform Resource Identifier for, this License with every copy or phonorecord of the Work You distribute, publicly display, publicly perform, or publicly digitally perform. You may not offer or impose any terms on the Work that alter or restrict the terms of this License or the recipients' exercise of the rights granted hereunder. You may not sublicense the Work. You must keep intact all notices that refer to this License and to the disclaimer of warranties. You may not distribute, publicly display, publicly perform, or publicly digitally perform the Work with any technological measures that control access or use of the Work in a manner inconsistent with the terms of this License Agreement. The above applies to the Work as incorporated in a Collective Work, but this does not require the Collective Work apart from the Work itself to be made subject to the terms of this License. If You create a Collective Work, upon notice from any Licensor You must, to the extent practicable, remove from the Collective Work any credit as required by clause 4(b), as requested. If You create a Derivative Work, upon notice from any Licensor You must, to the extent practicable, remove from the Derivative Work any credit as required by clause 4(b), as requested.

2. If you distribute, publicly display, publicly perform, or publicly digitally perform the Work or any Derivative Works or Collective Works, You must keep intact all copyright notices for the Work and provide, reasonable to the medium or means You are utilizing: (i) the name of the Original Author (or pseudonym, if applicable) if supplied, and/or (ii) if the Original Author and/or Licensor designate another party or parties (e.g. a sponsor institute, publishing entity, journal) for attribution in Licensor's copyright notice, terms of service or by other reasonable means, the name of such party or parties; the title of the Work if supplied; to the extent reasonably practicable, the Uniform Resource Identifier, if any, that Licensor specifies to be associated with the Work, unless such URI does not refer to the copyright notice or licensing information for the Work; and in the case of a Derivative Work, a credit identifying the use of the Work in the Derivative Work (e.g., "French translation of the Work by Original Author," or "Screenplay based on original Work by Original Author"). Such credit may be implemented in any reasonable manner; provided, however, that in the case of a Derivative Work or Collective Work, at a minimum such credit will appear where any other comparable authorship credit appears and in a manner at least as prominent as such other comparable authorship credit.

5. Representations, Warranties and Disclaimer

UNLESS OTHERWISE MUTUALLY AGREED TO BY THE PARTIES IN WRITING, LICENSOR OFFERS THE WORK AS-IS AND MAKES NO REPRESENTATIONS OR WARRANTIES OF ANY KIND CONCERNING THE WORK, EXPRESS, IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, WARRANTIES OF TITLE, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NONINFRINGEMENT, OR THE ABSENCE OF LATENT OR OTHER DEFECTS, ACCURACY, OR THE PRESENCE OF ABSENCE OF ERRORS, WHETHER OR NOT DISCOVERABLE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO SUCH EXCLUSION MAY NOT APPLY TO YOU.

6. Limitation on Liability. EXCEPT TO THE EXTENT REQUIRED BY APPLICABLE LAW, IN NO EVENT WILL LICENSOR BE LIABLE TO YOU ON ANY LEGAL THEORY FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES ARISING OUT OF THIS LICENSE OR THE USE OF THE WORK, EVEN IF LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH

DAMAGES.

7. Termination

1. This License and the rights granted hereunder will terminate automatically upon any breach by You of the terms of this License. Individuals or entities who have received Derivative Works or Collective Works from You under this License, however, will not have their licenses terminated provided such individuals or entities remain in full compliance with those licenses. Sections 1, 2, 5, 6, 7, and 8 will survive any termination of this License.

2. Subject to the above terms and conditions, the license granted here is perpetual (for the duration of the applicable copyright in the Work). Notwithstanding the above, Licensor reserves the right to release the Work under different license terms or to stop distributing the Work at any time; provided, however that any such election will not serve to withdraw this License (or any other license that has been, or is required to be, granted under the terms of this License), and this License will continue in full force and effect unless terminated as stated above.

8. Miscellaneous

1. Each time You distribute or publicly digitally perform the Work or a Collective Work, the Licensor offers to the recipient a license to the Work on the same terms and conditions as the license granted to You under this License.

2. Each time You distribute or publicly digitally perform a Derivative Work, Licensor offers to the recipient a license to the original Work on the same terms and conditions as the license granted to You under this License.

3. If any provision of this License is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this License, and without further action by the parties to this agreement, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

4. No term or provision of this License shall be deemed waived and no breach consented to unless such waiver or consent shall be in writing and signed by the party to be charged with such waiver or consent.

5. This License constitutes the entire agreement between the parties with respect to the Work licensed here. There are no understandings, agreements or representations with respect to the Work not specified here. Licensor shall not be bound by any additional provisions that may appear in any communication from You. This License may not be modified without the mutual written agreement of the Licensor and You.

1.295 logback-core 1.2.11

1.295.1 Available under license :

No license file was found, but licenses were detected in source scan.

/**

* Logback: the reliable, generic, fast and flexible logging framework.

* Copyright (C) 1999-2015, QOS.ch. All rights reserved.

*

* This program and the accompanying materials are dual-licensed under

* either the terms of the Eclipse Public License v1.0 as published by

* the Eclipse Foundation

- *
 - * or (per the licensee's choosing)
 - *
 - * under the terms of the GNU Lesser General Public License version 2.1
 - * as published by the Free Software Foundation.
- */

Found in path(s):

- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/ContextAwareImpl.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/ConsoleAppender.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/InterpretationContext.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BoldBlueCompositeConverter.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/ConverterUtil.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/FileNamePattern.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/InfoStatus.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ParamAction.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/SyslogAppenderBase.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/stax/EndEvent.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/EchoEncoder.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/ArchiveRemover.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/ConfigurableSSLServerSocketFactory.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/node/ComponentNode.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/LocationUtil.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/subst/Parser.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/PatternLayoutBase.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/FormattingNode.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/TimestampAction.java
- * /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/NonClosableInputStream.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/subst/Tokenizer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/ServerSocketListener.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/AutoFlushingObjectWriter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/AbstractEventEvaluatorAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/EndEvent.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/AppenderAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLConfigurable.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/FormattingConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/SyslogConstants.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/SSLServerSocketAppenderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLContextFactoryBean.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLConfigurableServerSocket.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/AbstractDiscriminator.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/AppenderAttachable.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/AbstractComponentTracker.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/PropertyContainer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/SiftingJoranConfiguratorBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/html/CssBuilder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/EnvUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/PeriodicityType.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/ServerRunner.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/RemoteReceiverClient.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/BasicStatusManager.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/SyslogOutputStream.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/SimpleKeywordNode.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/IfAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/AppenderFactory.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/RedCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/CharSequenceToRegexMapper.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/AppenderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/SizeBasedTriggeringPolicy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/LayoutWrappingEncoder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/ClientVisitor.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/Loader.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/PropertyDefinerBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/ReplacingCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/ANSIConstants.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/helpers/ThrowableToStringArray.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/CharSequenceState.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/DelayStrategy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/TrustManagerFactoryFactoryBean.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/Node.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/InPlayListener.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLConfiguration.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/StatusUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/recovery/ResilientSyslogOutputStream.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/DynamicClassLoadingException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/PropertySetterException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/SMTPAppenderBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ObjectWriterFactory.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/AppenderAttachableImpl.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/ServerListener.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/DefaultTimeBasedFileNamingAndTriggeringPolicy.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/layout/EchoLayout.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/DateTokenConverter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/HttpStatus.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/recovery/ResilientOutputStreamBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/FileUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/ComponentTracker.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLConfigurableSocket.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/util/IEscapeUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/FilterReply.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/ElementPath.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/IADDataForComplexProperty.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/LoginAuthenticator.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/FixedDelay.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/Appender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/StatusListenerAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/LayoutBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/AbstractSocketAppender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/SizeAndTimeBasedFNATP.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/html/NOPTrowableRenderer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/DatePatternToRegexUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/SocketConnector.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/Compiler.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/PatternLayoutEncoderBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ActionConst.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/DefaultSocketConnector.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/Parser.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/hook/ShutdownHookBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/util/ConfigurationWatchListUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/util/RegularEscapeUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/util/PropertySetter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/FileFilterUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/StatusBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/StatusManager.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/FilterAttachable.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/Discriminator.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/WarnStatus.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/hook/ShutdownHook.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/AbstractServerSocketAppender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/FormatInfo.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/CyclicBufferTracker.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/Duration.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/HostClassAndPropertyDouble.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/MagentaCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ShutdownHookAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/SystemInfo.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/KeyStoreFactoryBean.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/NopStatusListener.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/TimeUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/EventPlayer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/AbstractSSLSocketAppender.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/GreenCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/boolex/JaninoEventEvaluatorBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/LiteralConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/RemoteReceiverStreamClient.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/WhiteCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ObjectWriter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/MonoTypedConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/FilterAttachableImpl.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/CompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/RollingCalendar.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/DefaultDiscriminator.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/SiftingAppenderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/DefaultInvocationGate.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/FileSize.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BoldYellowCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/ContextBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/SimpleRuleStore.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/property/FileExistsPropertyDefiner.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/NewRuleAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/NoAutoStartUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/LifeCycleManager.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/subst/Node.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/ByteArrayUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/ScanException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/OnPrintStreamStatusListenerBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/SizeAndTimeBasedArchiveRemover.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/TriggeringPolicy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/stax/BodyEvent.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/boolex/Matcher.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/CyanCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/IdentityCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BoldWhiteCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/QueueFactory.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/SpacePadder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/DefinePropertyAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/JoranException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/DynamicConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/Interpreter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/OptionTokenizer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/filter/Filter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/StatusListenerConfigHelper.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/LogbackException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/filter/EvaluatorFilter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/html/IThrowableRenderer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/JNDIUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/BodyEvent.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/GrayCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/PostCompileProcessor.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/CoreConstants.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/StringCollectionUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/OutputStreamAppender.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/ConfigurableSSLSocketFactory.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/ConsoleTarget.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/Context.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/RollingFileAppender.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/TimeBasedArchiveRemover.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/util/StringToObjectConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/StatusListenerAsList.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/filter/AbstractMatcherFilter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/SaxEvent.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLParametersConfiguration.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ContextPropertyAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/LogbackLock.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/RolloverFailure.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/ContextUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/boolex/EventEvaluator.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/TokenConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/html/HTMLLayoutBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BlueCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/ForegroundCompositeConverterBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/recovery/RecoveryCoordinator.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/PropertyEvalScriptBuilder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/recovery/ResilientFileOutputStream.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/Condition.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/UnsynchronizedAppenderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/FixedWindowRollingPolicy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/TriggeringPolicyBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/util/AsIsEscapeUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/PropertyDefiner.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/stax/StaxEvent.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BoldGreenCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/read/ListAppender.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/RollingPolicyBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/Encoder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BlackCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/stax/StaxEventRecorder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/Status.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/XMLUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/TimeBasedFileNamingAndTriggeringPolicy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/PropertyWrapperForScripts.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ImplicitAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/AbstractAppenderFactoryUsingJoran.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/OnErrorConsoleStatusListener.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/OptionHelper.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ConversionRuleAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/helpers/CyclicBuffer.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/GenericConfigurator.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/stax/StartEvent.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSL.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/TimeBasedFileNamingAndTriggeringPolicyBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/ContextAware.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/ActionUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLComponent.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/subst/Token.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/CachingDateFormatter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/IncludeAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/ExecutorServiceUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/RemoteReceiverServerListener.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/NOPAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/DefaultNestedComponentRegistry.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/hook/DelayingShutdownHook.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/server/RemoteReceiverServerRunner.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/AppenderRefAction.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SSLNestedComponentRegistryRules.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/StartEvent.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/read/CyclicBufferAppender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/RollingPolicy.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/CompressionMode.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/IntegerTokenConverter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/ContextAwareBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/LifeCycle.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/FileStoreUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/CloseUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/helpers/Transform.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/ThenAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/StatusPrinter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/boolex/EvaluationException.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/StatusListener.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/util/AlmostAsIsEscapeUtil.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/FileAppender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/JoranConfiguratorBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/KeyManagerFactoryFactoryBean.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/Compressor.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/net/ssl/SecureRandomFactoryBean.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/ActionException.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/YellowCompositeConverter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/TimeBasedRollingPolicy.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/sift/AppenderTracker.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/color/BoldRedCompositeConverter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/PreSerializationTransformer.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/ElementSelector.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/AsyncAppenderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/NestedBasicPropertyIA.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/IADataForBasicProperty.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/Action.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/rolling/helper/RenameUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/action/NestedComplexPropertyIA.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/spi/DeferredProcessingAware.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/event/SaxEventRecorder.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/IncompatibleClassException.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/util/RestrictedEscapeUtil.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/encoder/EncoderBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/spi/NoAutoStart.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/Token.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/boolex/EventEvaluatorBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/AggregationType.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/joran/conditional/ThenOrElseActionBase.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/status/OnConsoleStatusListener.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/Converter.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/pattern/parser/TokenStream.java

* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-jar/ch/qos/logback/core/util/ContentTypeUtil.java


```
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/Layout.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/pattern/color/BoldCyanCompositeConverter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/joran/conditional/ElseAction.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/net/server/ConcurrentServerRunner.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/joran/spi/ConfigurationWatchList.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/pattern/parser/CompositeNode.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/helpers/NOPAppender.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/joran/spi/DefaultClass.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/property/ResourceExistsPropertyDefiner.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/status/ViewStatusMessagesServletBase.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/subst/NodeToStringTransformer.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/net/server/Client.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/pattern/color/BoldMagentaCompositeConverter.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/joran/spi/RuleStore.java
* /opt/cola/permits/1337069885_1654119262.0130525/0/logback-core-1-2-11-sources-3-
jar/ch/qos/logback/core/joran/action/PropertyAction.java
```

1.296 micronaut-sql-libraries 4.2.2

1.296.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/META-INF/native-
image/io.micronaut.sql/micronaut-jdbc/native-image.properties
```

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2020 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/JdbcDatabaseManager.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/BasicJdbcConfiguration.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/metadata/DataSourcePoolMetadataProvider.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/spring/SpringDataSourceResolver.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/spring/package-info.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/DataSourceResolver.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/spring/HibernatePresenceCondition.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/metadata/AbstractDataSourcePoolMetadata.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/spring/DataSourceTransactionManagerFactory.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/metadata/package-info.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/CalculatedSettings.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/package-info.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/metadata/DataSourcePoolMetadata.java
* /opt/cola/permits/1331473530_1653512647.1916578/0/micronaut-jdbc-4-2-2-sources-jar/io/micronaut/jdbc/nativeimage/JdbcFeature.java

1.297 pact-jvm-provider-junit5 4.0.10

1.297.1 Available under license :

Apache-2.0

1.298 javax.inject:javax.inject 1

1.298.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2009 The JSR-330 Expert Group
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Provider.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Named.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Qualifier.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Inject.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/package-info.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Scope.java
*/opt/cola/permits/1299411403_1650627395.55/0/javax-inject-1-sources-jar/javax/inject/Singleton.java
```

1.299 micronaut-flyway 5.2.0

1.299.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2021 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
```

* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/FlywayMigrator.java

No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2020 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/GormMigrationRunner.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/endpoint/FlywayEndpoint.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/DataSourceMigrationRunner.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/event/SchemaCleanedEvent.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/graalvm/FeatureDetectorSubstitutions.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/endpoint/FlywayReport.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/AbstractFlywayMigration.java

* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-

```
jar/io/micronaut/flyway/event/MigrationFinishedEvent.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/graalvm/FlywayFeature.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/graalvm/ScannerSubstitutions.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/package-info.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/FlywayConfigurationProperties.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/graalvm/MicronautPathLocationScanner.java
* /opt/cola/permits/1331473697_1653068637.3698914/0/micronaut-flyway-5-2-0-sources-
jar/io/micronaut/flyway/AlternativeMigrationRunner.java
```

1.300 feign-jackson 8.18.0

1.300.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
* Copyright 2013 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* http://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1274699876_1645234767.83/0/feign-jackson-8-18-0-sources-
jar/feign/jackson/JacksonEncoder.java
* /opt/cola/permits/1274699876_1645234767.83/0/feign-jackson-8-18-0-sources-
jar/feign/jackson/JacksonDecoder.java
```

1.301 kindedj 1.1.0

1.301.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or

agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2017 KindedJ

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.302 micronaut-kotlin-integrations 3.2.2

1.302.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/KtorApplicationBuilder.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/KtorApplicationBuilder.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/server/AbstractKtorEmbeddedServer.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/KtorRoutingBuilder.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/server/KtorTomcatEmbeddedServer.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/env/MicronautKtorEnvironmentConfig.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/server/KtorNettyEmbeddedServer.kt
```

* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/factory/MicronautKtorApplicationFactory.kt
* /opt/cola/permits/1331473908_1653514800.6790595/0/micronaut-ktor-3-2-2-sources-jar/io/micronaut/ktor/server/KtorJettyEmbeddedServer.kt

1.303 kotlin-stdlib-common 1.3.71

1.303.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications

or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions

stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

1.304 cloudevents---api 2.1.1

1.304.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2018-Present The CloudEvents Authors
~ <p>
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~ <p>
~ http://www.apache.org/licenses/LICENSE-2.0
```

~ <p>
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
~
-->

Found in path(s):

* /opt/cola/permits/1337069596_1654221635.0898836/0/cloudevents-api-2-1-1-jar/META-INF/maven/io.cloudevents/cloudevents-api/pom.xml

1.305 metrics-integration-for-logback 4.1.17

1.305.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0
Automatic-Module-Name: com.codahale.metrics.logback
Build-Jdk-Spec: 1.8
Bundle-Description: An instrumented appender for Logback.
Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>
Bundle-ManifestVersion: 2
Bundle-Name: Metrics Integration for Logback
Bundle-SymbolicName: io.dropwizard.metrics.logback
Bundle-Version: 4.1.17
Created-By: Apache Maven Bundle Plugin
Export-Package: com.codahale.metrics.logback;uses:="ch.qos.logback.classic.spi,ch.qos.logback.core,com.codahale.metrics";version="4.1.17"
Implementation-Title: Metrics Integration for Logback
Implementation-Version: 4.1.17
Import-Package: ch.qos.logback.classic;version="[1.2,2)",ch.qos.logback.classic.spi;version="[1.2,2)",ch.qos.logback.core;version="[1.2,2)",com.codahale.metrics;version="[4.1,5)"
Private-Package: com.codahale.metrics.logback
Require-Capability: osgi.ee;filter:="(&(osgi.ee=JavaSE)(version=1.8))"
Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188427_1654734186.039732/0/metrics-logback-4-1-17-jar/META-INF/MANIFEST.MF

1.306 micronaut-cache 3.3.0

1.306.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/jcache/JCacheSyncCache.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/DefaultCacheErrorHandler.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/serialize/package-info.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/interceptor/DefaultCacheKeyGenerator.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/Cache.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/CacheInfo.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/package-info.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/annotation/CacheInvalidate.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/CacheErrorHandler.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/AsyncCacheErrorHandler.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/annotation/package-info.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/interceptor/ParametersKey.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/annotation/PutOperations.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/interceptor/CacheKeyGenerator.java
```

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/discovery/DiscoveryClientCacheConfiguration.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/jcache/package-info.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/interceptor/CacheInterceptor.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/interceptor/KotlinSuspendFunCacheKeyGenerator.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/annotation/Cacheable.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/SyncCache.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/discovery/package-info.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/exceptions/package-info.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/AbstractMapBasedSyncCache.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/serialize/DefaultStringKeySerializer.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/DelegatingAsyncCache.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/annotation/InvalidateOperations.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/DynamicCacheManager.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/jcache/JCacheManager.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/DefaultCacheManager.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/AsyncCache.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/interceptor/package-info.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/discovery/CachingCompositeDiscoveryClient.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/exceptions/CacheSystemException.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/jcache/metrics/JCacheMetricsBinder.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/annotation/CacheConfig.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/annotation/CachePut.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/CacheConfiguration.java

* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/io/micronaut/cache/DelegatingAsyncBlockingCache.java


```
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/interceptor/ValueSupplierException.java
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/CacheManager.java
No license file was found, but licenses were detected in source scan.
```

```
/*
* Copyright 2017-2021 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* https://www.apache.org/licenses/LICENSE-2.0
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/
```

Found in path(s):

```
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-
jar/io/micronaut/cache/annotation/CacheAnnotation.java
No license file was found, but licenses were detected in source scan.
```

```
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
# distributed under the License is distributed on an "AS IS" BASIS,
```

Found in path(s):

```
* /opt/cola/permits/1331473554_1653513966.161154/0/micronaut-cache-core-3-3-0-sources-jar/META-INF/native-
image/io.micronaut.cache/micronaut-cache-core/native-image.properties
```

1.307 micronaut-aws 3.2.1

1.307.1 Available under license :

Apache License
Version 2.0, January 2004
<https://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.308 log4j-over-slf4j 1.7.30

1.308.1 Available under license :

No license file was found, but licenses were detected in source scan.

<url><http://www.apache.org/licenses/LICENSE-2.0.txt></url>

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/META-INF/maven/org.slf4j/log4j-over-slf4j/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/helpers/NullEnumeration.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2001-2004 The Apache Software Foundation.

*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,

- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/Configurator.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/LoggerFactory.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/NDC.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Appender.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/BasicConfigurator.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/ConsoleAppender.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/LoggerRepository.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Layout.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/WriterAppender.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/xml/DOMConfigurator.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/SimpleLayout.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/LoggingEvent.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/RollingFileAppender.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/HierarchyEventListener.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/PatternLayout.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Logger.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Level.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Category.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Priority.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/ErrorHandler.java
- * /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/OptionHandler.java

* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/FileAppender.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/LogManager.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/AppenderSkeleton.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/helpers/LogLog.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/PropertyConfigurator.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/Log4jLoggerFactory.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/MDC.java
* /opt/ws_local/PERMITS_SQL/1088524438_1599802440.27/0/log4j-over-slf4j-1-7-30-sources-1-jar/org/apache/log4j/spi/Filter.java

1.309 rxjersey 0.9.0

1.309.1 Available under license :

MIT License

Copyright (c) 2016 Alex Shpak

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.310 rxjava 2.2.21

1.310.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or

Derivative Works a copy of this License; and

- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.311 okio 2.8.0

1.311.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2017 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Utf8.kt

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2018 Square, Inc.
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
```

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/-
DeprecatedUpgrade.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/-
DeprecatedUtf8.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-
jar/commonMain/okio/PeekSource.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/-
DeprecatedOkio.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Throttler.kt
No license file was found, but licenses were detected in source scan.

/*

* Licensed to the Apache Software Foundation (ASF) under one or more
* contributor license agreements. See the NOTICE file distributed with
* this work for additional information regarding copyright ownership.
* The ASF licenses this file to You under the Apache License, Version 2.0
* (the "License"); you may not use this file except in compliance with
* the License. You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/-Base64.kt
No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2014 Square, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/SegmentPool.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Sink.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Source.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/BufferedSource.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/BufferedSink.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Buffer.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Timeout.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/SegmentPool.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/ForwardingSource.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/JvmOkio.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/InflaterSource.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Segment.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/GzipSink.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/DeflaterSink.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/AsyncTimeout.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/RealBufferedSource.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/ForwardingSink.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/GzipSource.kt
- * /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/RealBufferedSink.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2015 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/SegmentedByteString.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/ForwardingTimeout.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/SegmentedByteString.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2018 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/ByteString.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/-Util.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/-Platform.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/-Platform.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/ByteString.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/Utf8.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2016 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/HashingSink.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/Pipe.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/HashingSource.kt
* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Options.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 Square Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/jvmMain/okio/ByteString.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/SegmentedByteString.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/RealBufferedSink.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/RealBufferedSource.kt

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2019 Square, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Okio.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/internal/Buffer.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/BufferedSource.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/BufferedSink.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Timeout.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Source.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/RealBufferedSink.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/RealBufferedSource.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Buffer.kt

* /opt/cola/permits/1340031362_1654618471.6843367/0/okio-2-8-0-sources-2-jar/commonMain/okio/Sink.kt

1.312 jersey-container-servlet-core 2.32

1.312.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

* License MIT (<http://www.opensource.org/licenses/mit-license.php>)

* Project: <http://angularjs.org>

* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

* License: all the source code provided by AOP Alliance is Public Domain.

* Project: <http://aopalliance.sourceforge.net>

* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

* License: Apache License, 2.0

* Project: <http://beanvalidation.org/1.1/>

* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors

* by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

* License: Apache License, 2.0

* Project: <https://beanvalidation.org/>

* Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7

* License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

* Project: <http://getbootstrap.com>

* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

* License: Apache License, 2.0

* Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

* License: Apache License, 2.0

* Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

* License: Apache License, 2.0

* Project: <http://www.javassist.org/>

* Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

* License: jquery.org/license

* Project: jquery.org

* Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

* License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)

* Project: <http://www.pasella.it/projects/jquery/barcode>

* Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

* License: CC0

* No copyright

* Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/>
- # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which

are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall

apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness

for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged

intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of

the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s),

version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis

or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other

recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not

signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.313 feign-core 8.18.0

1.313.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2015 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Param.java
* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/QueryMap.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2014 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/InvocationHandlerFactory.java
* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/SynchronousMethodHandler.java
No license file was found, but licenses were detected in source scan.

/*
* Copyright (C) 2008 Google Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Types.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Response.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codecs/ErrorDecoder.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Contract.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codecs/EncodeException.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Request.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/RequestInterceptor.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/auth/Base64.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Util.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/RequestTemplate.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Client.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codecs/StringDecoder.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codecs/DecodeException.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/FeignException.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Target.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-

jar/feign/auth/BasicAuthRequestInterceptor.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/RetryableException.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Logger.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/ReflectiveFeign.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Retryer.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/Feign.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/MethodMetadata.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codec/Decoder.java

* /opt/cola/permits/1274700542_1645235077.45/0/feign-core-8-18-0-sources-jar/feign/codec/Encoder.java

1.314 commons-codec 1.15

1.314.1 Available under license :

Apache Commons Codec

Copyright 2002-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<https://www.apache.org/>).

src/test/org/apache/commons/codec/language/DoubleMetaphoneTest.java

contains test data from <http://aspell.net/test/orig/batch0.tab>.

Copyright (C) 2002 Kevin Atkinson (kevina@gnu.org)

=====
The content of package org.apache.commons.codec.language.bm has been translated from the original php source code available at <http://stevemorse.org/phoneticinfo.htm> with permission from the original authors.

Original source copyright:

Copyright (c) 2008 Alexander Beider & Stephen P. Morse.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.315 jackson-bom 2.13.2

1.315.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a

copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct

or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of

this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following

boilerplate notice, with the fields enclosed by brackets "[]"
replaced with your own identifying information. (Don't include
the brackets!) The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same "printed page" as the copyright notice for easier
identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.316 kotlin-libraries-bill-of-materials 1.6.10

1.316.1 Available under license :

The version of Rhino used in GWT is licensed under a dual license,
Netscape Public License 1.1 / GNU General Public License.

The text of the Netscape Public License

is provided below (<http://website-archive.mozilla.org/www.mozilla.org/mpl/MPL/NPL/1.1/>):

AMENDMENTS

The Netscape Public License Version 1.1 ("NPL") consists of the
Mozilla Public License Version 1.1 with the following Amendments,
including Exhibit A-Netscape Public License. Files identified with
"Exhibit A-Netscape Public License" are governed by the Netscape
Public License Version 1.1.

Additional Terms applicable to the Netscape Public License.

I. Effect.

These additional terms described in this Netscape Public
License -- Amendments shall apply to the Mozilla Communicator
client code and to all Covered Code under this License.

II. "Netscape's Branded Code" means Covered Code that Netscape

distributes and/or permits others to distribute under one or more trademark(s) which are controlled by Netscape but which are not licensed for use under this License.

III. Netscape and logo.

This License does not grant any rights to use the trademarks "Netscape", the "Netscape N and horizon" logo or the "Netscape lighthouse" logo, "Netcenter", "Gecko", "Java" or "JavaScript", "Smart Browsing" even if such marks are included in the Original Code or Modifications.

IV. Inability to Comply Due to Contractual Obligation.

Prior to licensing the Original Code under this License, Netscape has licensed third party code for use in Netscape's Branded Code. To the extent that Netscape is limited contractually from making such third party code available under this License, Netscape may choose to reintegrate such code into Covered Code without being required to distribute such code in Source Code form, even if such code would otherwise be considered "Modifications" under this License.

V. Use of Modifications and Covered Code by Initial Developer.

V.1. In General.

The obligations of Section 3 apply to Netscape, except to the extent specified in this Amendment, Section V.2 and V.3.

V.2. Other Products.

Netscape may include Covered Code in products other than the Netscape's Branded Code which are released by Netscape during the two (2) years following the release date of the Original Code, without such additional products becoming subject to the terms of this License, and may license such additional products on different terms from those contained in this License.

V.3. Alternative Licensing.

Netscape may license the Source Code of Netscape's Branded Code, including Modifications incorporated therein, without such Netscape Branded Code becoming subject to the terms of this License, and may license such Netscape Branded Code on different terms from those

contained in this License.

VI. Litigation.

Notwithstanding the limitations of Section 11 above, the provisions regarding litigation in Section 11(a), (b) and (c) of the License shall apply to all disputes relating to this License.

EXHIBIT A-Netscape Public License.

"The contents of this file are subject to the Netscape Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Mozilla Communicator client code, released March 31, 1998.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1998-1999 Netscape Communications Corporation. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [____] License and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the [____] License."

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Code, prior Modifications used by a Contributor, and the Modifications made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Original Code and Modifications, in each case including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally accepted in the software development community for the electronic transfer of data.

1.5. "Executable" means Covered Code in any form other than Source Code.

1.6. "Initial Developer" means the individual or entity identified as the Initial Developer in the Source Code notice required by Exhibit A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications

made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating

to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due

to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such

Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers) which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as Multiple-Licensed. Multiple-Licensed means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.
Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the [_____] License), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete the provisions above, a recipient may use your version of this file under either the MPL or the [_____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

=====

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to

using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below,

refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include

anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.>

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.
This is free software, and you are welcome to redistribute it
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the program
`Gnomovision' (which makes passes at compilers) written by James Hacker.
```

```
<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License.

=====

SDL 2.0 and newer are available under the zlib license :

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software

in a product, an acknowledgment in the product documentation would be appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1996 - 2021, Daniel Stenberg, <daniel@haxx.se>, and many contributors, see the THANKS file.

All rights reserved.

Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

=====
libc++ License
=====

The libc++ library is dual licensed under both the University of Illinois "BSD-Like" license and the MIT license. As a user of this code you may choose to use it under either license. As a contributor, you agree to allow your code to be used under both.

Full text of the relevant licenses is included below.

=====
University of Illinois/NCSA
Open Source License

Copyright (c) 2009-2017 by the contributors listed in CREDITS.TXT

All rights reserved.

Developed by:

LLVM Team

University of Illinois at Urbana-Champaign

<http://llvm.org>

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal with the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimers.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimers in the documentation and/or other materials provided with the distribution.
- * Neither the names of the LLVM Team, University of Illinois at Urbana-Champaign, nor the names of its contributors may be used to endorse or promote products derived from this Software without specific prior written permission.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE CONTRIBUTORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS WITH THE SOFTWARE.

=====

Copyright (c) 2009-2014 by the contributors listed in CREDITS.TXT

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the

Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the

same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

=====
=====

APACHE HARMONY SUBCOMPONENTS:

Apache Harmony includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

License Notice for ICU4C version 3.4

=====

ICU License - ICU 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2005 International Business Machines Corporation and others
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons

to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

=====
=====

License Notice for ICU4J version 4.2.1

=====

ICU4J license - ICU4J 1.8.1 and later

COPYRIGHT AND PERMISSION NOTICE

Copyright (c) 1995-2009 International Business Machines Corporation and others

All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, provided that the above copyright notice(s) and this permission notice appear

in all copies of the Software and that both the above copyright notice(s) and this permission notice appear in supporting documentation.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Software without prior written authorization of the copyright holder.

All trademarks and registered trademarks mentioned herein are the property of their respective owners.

=====
=====

License Notice for FDLIBM version 5.2
=====

Copyright (C) 2004 by Sun Microsystems, Inc. All rights reserved.

Developed at SunSoft, a Sun Microsystems, Inc. business.
Permission to use, copy, modify, and distribute this software is freely granted, provided that this notice is preserved.

=====
=====

License Notice for ZLIB version 1.2.3
=====

(C) 1995-2004 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

=====
=====

Visual C++ Runtime Files

=====

The file msvc71.dll is the Microsoft(R) C Runtime Library.
The file msvcp71.dll is the Microsoft Multithreaded Standard C++ Library.

These files are redistributed from Microsoft(R) Visual Studio 7.1 as described in the file "redist.txt" available with that product.

=====
=====

License Notice for Bouncy Castle version 1.45

=====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
=====

License Notice for MX4J 3.0.2.

=====

The MX4J License, Version 1.0

Copyright (c) 2001-2004 by the MX4J contributors. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
"This product includes software developed by the

MX4J project (<http://mx4j.sourceforge.net>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

- 4. The name "MX4J" must not be used to endorse or promote products derived from this software without prior written permission.

For written permission, please contact
biorn_steedom [at] users [dot] sourceforge [dot] net

- 5. Products derived from this software may not be called "MX4J", nor may "MX4J" appear in their name, without prior written permission of Simone Bordet.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE MX4J CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====

This software consists of voluntary contributions made by many individuals on behalf of the MX4J project. For more information on MX4J, please see the MX4J website (<http://mx4j.sourceforge.net/>).

=====

Notice for The Independent JPEG Group's JPEG software

=====

This distribution may contain software that is based in part on the work of the Independent JPEG Group for which the following notice applies:

The authors make NO WARRANTY or representation, either express or implied, with respect to this software, its quality, accuracy, merchantability, or fitness for a particular purpose. This software is provided "AS IS", and you,

its user, assume the entire risk as to its quality and accuracy.

This software is copyright (C) 1991-1998, Thomas G. Lane.
All Rights Reserved except as specified below.

Permission is hereby granted to use, copy, modify, and distribute this software (or portions thereof) for any purpose, without fee, subject to these conditions:

- (1) If any part of the source code for this software is distributed, then this README file must be included, with this copyright and no-warranty notice unaltered; and any additions, deletions, or changes to the original files must be clearly indicated in accompanying documentation.
- (2) If only executable code is distributed, then the accompanying documentation must state that "this software is based in part on the work of the Independent JPEG Group".
- (3) Permission for use of this software is granted only if the user accepts full responsibility for any undesirable consequences; the authors accept NO LIABILITY for damages of any kind.

These conditions apply to any software derived from or based on the IJG code, not just to the unmodified library. If you use our work, you ought to acknowledge us.

Permission is NOT granted for the use of any IJG author's name or company name in advertising or publicity relating to this software or products derived from it. This software may be referred to only as "the Independent JPEG Group's software".

We specifically permit and encourage the use of this software as the basis of commercial products, provided that all warranty or liability claims are assumed by the product vendor.

=====
=====

Notice for Little CMS

=====

This distribution may contain software that is based in part on Little cms for which the following notice applies:

Copyright (c) 1998-2007 Marti Maria Saguer

Permission is hereby granted, free of charge, to any person obtaining a copy

of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

=====
=====

Notice for libpng

=====

This distribution may contain software that is based in part on libpng v1.2.38 for which the following notice applies:

libpng versions 1.2.6, August 15, 2004, through 1.2.38, July 16, 2009, are Copyright (c) 2004, 2006-2009 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.2.5 with the following individual added to the list of Contributing Authors:

Cosmin Truta

libpng versions 1.0.7, July 1, 2000, through 1.2.5, October 3, 2002, are Copyright (c) 2000-2002 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-1.0.6 with the following individuals added to the list of Contributing Authors:

Simon-Pierre Cadieux
Eric S. Raymond
Gilles Vollant

and with the following additions to the disclaimer:

There is no warranty against interference with your enjoyment of the library or against infringement. There is no warranty that our efforts or the library will fulfill any of your particular purposes or needs. This library is provided with all faults, and the entire risk of satisfactory quality, performance, accuracy, and effort is with the user.

libpng versions 0.97, January 1998, through 1.0.6, March 20, 2000, are Copyright (c) 1998, 1999, 2000 Glenn Randers-Pehrson, and are distributed according to the same disclaimer and license as libpng-0.96, with the following individuals added to the list of Contributing Authors:

Tom Lane
Glenn Randers-Pehrson
Willem van Schaik

libpng versions 0.89, June 1996, through 0.96, May 1997, are Copyright (c) 1996, 1997 Andreas Dilger
Distributed according to the same disclaimer and license as libpng-0.88, with the following individuals added to the list of Contributing Authors:

John Bowler
Kevin Bracey
Sam Bushell
Magnus Holmgren
Greg Roelofs
Tom Tanner

libpng versions 0.5, May 1995, through 0.88, January 1996, are Copyright (c) 1995, 1996 Guy Eric Schalnat, Group 42, Inc.

For the purposes of this copyright and license, "Contributing Authors" is defined as the following set of individuals:

Andreas Dilger
Dave Martindale
Guy Eric Schalnat
Paul Schmidt
Tim Wegner

The PNG Reference Library is supplied "AS IS". The Contributing Authors and Group 42, Inc. disclaim all warranties, expressed or implied, including, without limitation, the warranties of merchantability and of fitness for any purpose. The Contributing Authors and Group 42, Inc. assume no liability for direct, indirect, incidental, special, exemplary, or consequential damages, which may result from the use of the PNG Reference Library, even if advised of the possibility of such damage.

Permission is hereby granted to use, copy, modify, and distribute this source code, or portions hereof, for any purpose, without fee, subject to the following restrictions:

1. The origin of this source code must not be misrepresented.
2. Altered versions must be plainly marked as such and must not be misrepresented as being the original source.
3. This Copyright notice may not be removed or altered from any source or altered source distribution.

The Contributing Authors and Group 42, Inc. specifically permit, without fee, and encourage the use of this source code as a component to supporting the PNG file format in commercial products. If you use this source code in a product, acknowledgment is not required but would be appreciated.

=====
=====

Notice for Unicode Character Database

=====

Copyright (c) 1991-2005 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>.

Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF THIRD

PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

=====
=====

Notice for IETF RFCs prior to 2005

=====

This distribution may contain software that is based in part on the work of the following IETF RFCs. Portions of the IETF RFC specifications may be included in source code comments for reference, and in accordance with the licensing terms, the licenses are reproduced here:

RFC 1779 - A String Representation of Distinguished Names

RFC 2045 - Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies

RFC 2251 - Lightweight Directory Access Protocol (v3)
Copyright (C) The Internet Society (1997). All Rights Reserved.

RFC 2253 - Lightweight Directory Access Protocol (v3): UTF-8 String Representation of Distinguished Names
Copyright (C) The Internet Society (1997). All Rights Reserved.

RFC 2313 - PKCS #1: RSA Encryption
Copyright (C) The Internet Society (1998). All Rights Reserved.

RFC 2315 - PKCS #7: Cryptographic Message Syntax
Copyright (C) The Internet Society (1998). All Rights Reserved.

RFC 2459 - Internet X.509 Public Key Infrastructure Certificate and CRL Profile
Copyright (C) The Internet Society (1999). All Rights Reserved.

RFC 2616 - Hypertext Transfer Protocol -- HTTP/1.1

Copyright (C) The Internet Society (1999). All Rights Reserved.

RFC 2781 - UTF-16, an encoding of ISO 10646

Copyright (C) The Internet Society (2000). All Rights Reserved.

RFC 2891 - LDAP Control Extension for Server Side Sorting of Search Results

Copyright (C) The Internet Society (2000). All Rights Reserved.

RFC 2985 - PKCS #9: Selected Object Classes and Attribute Types

Copyright (C) The Internet Society (2000). All Rights Reserved.

RFC 3161 - Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)

Copyright (C) The Internet Society (2001). All Rights Reserved.

RFC 3280 - Internet X.509 Public Key Infrastructure, Certificate and
Certificate Revocation List (CRL) Profile

Copyright (C) The Internet Society (2002). All Rights Reserved.

INTERNET-DRAFT - Persistent Search: A Simple LDAP Change Notification Mechanism

<draft-ietf-ldapext-psearch-03.txt>

Copyright (C) The Internet Society (1997-2000). All Rights Reserved.

The following statement applies to each RFC:

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Notice for IETF RFCs from 2005 onwards

=====

This distribution may contain software that is based in part on the work of the following IETF RFCs. Portions of the IETF RFC specifications may be included in source code comments for reference, and in accordance with the licensing terms, the licenses are reproduced here:

RFC 4122 - A Universally Unique Identifier (UUID) URN Namespace
Copyright (C) The Internet Society (2005).

The following statement applies to this RFC:

Copyright (C) The Internet Society (2005).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

=====

Notice for RSA Public-Key Cryptography Standards

=====

Portions of Apache Harmony reference RSA Public-Key Cryptography Standards: PKCS#7, PKCS#8 and PKSC#10. Portions of these standards are included in Harmony Javadoc for reference, and in accordance with the licensing terms for PKCS#7, PKCS#8 and PKSC#10, the full copyright statement is here:

Copyright 1991-1993 RSA Laboratories, a division of RSA Data Security, Inc.
License to copy this document is granted provided that it is identified as

"RSA Data Security, Inc. Public-Key Cryptography Standards (PKCS)" in all material mentioning or referencing this document.

=====
=====

Notice for Apache Yoko

=====

This distribution may contain software that is based in part on Apache Yoko for which the following notice applies:

Apache Yoko is an effort undergoing incubation at the Apache Software Foundation (ASF). Incubation is required of all newly accepted projects until a further review indicates that the infrastructure, communications, and decision making process have stabilized in a manner consistent with other successful ASF projects. While incubation status is not necessarily a reflection of the completeness or stability of the code, it does indicate that the project has yet to be fully endorsed by the ASF.

<http://incubator.apache.org/yoko/>

=====
=====

Notice for the Eclipse JDT Core Batch Compiler

=====

The license terms for the Eclipse JDT Core Batch Compiler are available at <http://www.eclipse.org/legal/epl-v10.html>

Source code for the Eclipse JDT Core Batch Compiler can be found at: <http://download.eclipse.org/eclipse/downloads/>

Navigate to the version of ECJ used in this distribution, and the source is provided in a ZIP file in the section titled "JDT Core Batch Compiler".

=====
=====

License notice for the DejaVu fonts package
=====

Fonts are (c) Bitstream (see below). DejaVu changes are in public domain.
Glyphs imported from Arev fonts are (c) Tavmjung Bah (see below)

Bitstream Vera Fonts Copyright

Copyright (c) 2003 by Bitstream, Inc. All Rights Reserved. Bitstream Vera is
a trademark of Bitstream, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy
of the fonts accompanying this license ("Fonts") and associated
documentation files (the "Font Software"), to reproduce and distribute the
Font Software, including without limitation the rights to use, copy, merge,
publish, distribute, and/or sell copies of the Font Software, and to permit
persons to whom the Font Software is furnished to do so, subject to the
following conditions:

The above copyright and trademark notices and this permission notice shall
be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular
the designs of glyphs or characters in the Fonts may be modified and
additional glyphs or characters may be added to the Fonts, only if the fonts
are renamed to names not containing either the words "Bitstream" or the word
"Vera".

This License becomes null and void to the extent applicable to Fonts or Font
Software that has been modified and is distributed under the "Bitstream
Vera" names.

The Font Software may be sold as part of a larger software package but no
copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT,
TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL BITSTREAM OR THE GNOME
FOUNDATION BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING
ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES,
WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF

THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the names of Gnome, the Gnome Foundation, and Bitstream Inc., shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from the Gnome Foundation or Bitstream Inc., respectively. For further information, contact: fonts at gnome dot org.

Arev Fonts Copyright

Copyright (c) 2006 by Tavmjong Bah. All Rights Reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of the fonts accompanying this license ("Fonts") and associated documentation files (the "Font Software"), to reproduce and distribute the modifications to the Bitstream Vera Font Software, including without limitation the rights to use, copy, merge, publish, distribute, and/or sell copies of the Font Software, and to permit persons to whom the Font Software is furnished to do so, subject to the following conditions:

The above copyright and trademark notices and this permission notice shall be included in all copies of one or more of the Font Software typefaces.

The Font Software may be modified, altered, or added to, and in particular the designs of glyphs or characters in the Fonts may be modified and additional glyphs or characters may be added to the Fonts, only if the fonts are renamed to names not containing either the words "Tavmjong Bah" or the word "Arev".

This License becomes null and void to the extent applicable to Fonts or Font Software that has been modified and is distributed under the "Tavmjong Bah Arev" names.

The Font Software may be sold as part of a larger software package but no copy of one or more of the Font Software typefaces may be sold by itself.

THE FONT SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OF COPYRIGHT, PATENT, TRADEMARK, OR OTHER RIGHT. IN NO EVENT SHALL TAVMJONG BAH BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, INCLUDING ANY GENERAL, SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL

DAMAGES, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF THE USE OR INABILITY TO USE THE FONT SOFTWARE OR FROM OTHER DEALINGS IN THE FONT SOFTWARE.

Except as contained in this notice, the name of Tavmjong Bah shall not be used in advertising or otherwise to promote the sale, use or other dealings in this Font Software without prior written authorization from Tavmjong Bah. For further information, contact: tavmjong @ free . fr.

=====
=====

License notice for ASM

=====

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====
=====

License notice for Junit
=====

This distribution may contain a jar file based on Junit 4.2.1 for which the following notice applies:

Copyright 2001-2009 Kent Beck <kent@threeiversinstitute.org>
Copyright 2000-2009 Erich Gamma <erich_gamma@acm.org>
Copyright 2006-2009 David Saff <david@saff.net>

Common Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product

offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. IBM is the initial Agreement Steward. IBM may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

=====
=====

The Apache Harmony DRLVM uses the Apache Portable Runtime which includes a number of subcomponents with separate copyright notices and license terms. Your use of the source code for these subcomponents is subject to the terms and conditions of the following licenses.

From strings/apr_fnmatch.c, include/apr_fnmatch.h, misc/unix/getopt.c, file_io/unix/mktemp.c, strings/apr_strings.c:

```
/*
 * Copyright (c) 1987, 1993, 1994
 *   The Regents of the University of California. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 *   notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *   notice, this list of conditions and the following disclaimer in the
 *   documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *   must display the following acknowledgement:
 *   This product includes software developed by the University of
 *   California, Berkeley and its contributors.
 * 4. Neither the name of the University nor the names of its contributors
 *   may be used to endorse or promote products derived from this software
 *   without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
```

From network_io/unix/inet_ntop.c, network_io/unix/inet_pton.c:

```
/* Copyright (c) 1996 by Internet Software Consortium.
 *
 * Permission to use, copy, modify, and distribute this software for any
 * purpose with or without fee is hereby granted, provided that the above
```

* copyright notice and this permission notice appear in all copies.
*
* THE SOFTWARE IS PROVIDED "AS IS" AND INTERNET SOFTWARE CONSORTIUM DISCLAIMS
* ALL WARRANTIES WITH REGARD TO THIS SOFTWARE INCLUDING ALL IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS. IN NO EVENT SHALL INTERNET SOFTWARE
* CONSORTIUM BE LIABLE FOR ANY SPECIAL, DIRECT, INDIRECT, OR CONSEQUENTIAL
* DAMAGES OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR
* PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS
* ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THIS
* SOFTWARE.

From dso/aix/dso.c:

* Based on libdl (dlfcn.c/dlfcn.h) which is
* Copyright (c) 1992,1993,1995,1996,1997,1988
* Jens-Uwe Mager, Helios Software GmbH, Hannover, Germany.
*
* Not derived from licensed software.
*
* Permission is granted to freely use, copy, modify, and redistribute
* this software, provided that the author is not construed to be liable
* for any results of using the software, alterations are clearly marked
* as such, and this notice is not modified.

From strings/apr_strnatcmp.c, include/apr_strings.h:

strnatcmp.c -- Perform 'natural order' comparisons of strings in C.
Copyright (C) 2000 by Martin Pool <mbp@humblebug.org.au>

This software is provided 'as-is', without any express or implied
warranty. In no event will the authors be held liable for any damages
arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,
including commercial applications, and to alter it and redistribute it
freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not
claim that you wrote the original software. If you use this software
in a product, an acknowledgment in the product documentation would be
appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be
misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

From strings/apr_snprintf.c:

*

* cvt - IEEE floating point formatting routines.
* Derived from UNIX V7, Copyright(C) Caldera International Inc.
*

Copyright(C) Caldera International Inc. 2001-2002. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code and documentation must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

All advertising materials mentioning features or use of this software must display the following acknowledgement:

This product includes software developed or owned by Caldera International, Inc.

Neither the name of Caldera International, Inc. nor the names of other contributors may be used to endorse or promote products derived from this software without specific prior written permission.

USE OF THE SOFTWARE PROVIDED FOR UNDER THIS LICENSE BY CALDERA INTERNATIONAL, INC. AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL CALDERA INTERNATIONAL, INC. BE LIABLE FOR ANY DIRECT, INDIRECT INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====
=====

License notice for Hamcrest

=====
BSD License

Copyright (c) 2000-2006, www.hamcrest.org
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of
conditions and the following disclaimer. Redistributions in binary form must reproduce
the above copyright notice, this list of conditions and the following disclaimer in
the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse
or promote products derived from this software without specific prior written
permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND
ANY
EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO
EVENT
SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED
TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN
CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN
ANY
WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH
DAMAGE.

=====
=====
The Trove library is licensed under the Lesser GNU Public License,
which is included with the distribution in a file called trove_license.txt.

The PrimeFinder and HashFunctions classes in Trove are subject to the
following license restrictions:

Copyright (c) 1999 CERN - European Organization for Nuclear Research.

Permission to use, copy, modify, distribute and sell this software and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation. CERN makes no representations about the suitability of this software for any purpose. It is provided "as is" without expressed or implied warranty.

Copyright (c) 2005-2010 Sam Stephenson

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright (C) 2009-2015 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

MIT License

Copyright (c) 2019 Microsoft Corporation, Daan Leijen

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
=====
== NOTICE file corresponding to the section 4 d of           ==
== the Apache License, Version 2.0,                          ==
== in this case for the Kotlin Compiler distribution.         ==
=====
```

Kotlin Compiler
Copyright 2010-2020 JetBrains s.r.o and respective authors and developers
Eclipse Distribution License - v 1.0

Copyright (c) 2007, Eclipse Foundation, Inc. and its licensors.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN

IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Eclipse Public License - v 1.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no

assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under its own license agreement, provided that:

a) it complies with the terms and conditions of this Agreement; and

b) its license agreement:

i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential

liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such

litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

This license applies to all parts of Dart that are not externally maintained libraries. The external maintained libraries used by Dart are:

7-Zip - in third_party/7zip
JSCORE - in runtime/third_party/jscre
Ant - in third_party/apache_ant
args4j - in third_party/args4j
bzip2 - in third_party/bzip2
dromaeo - in samples/third_party/dromaeo
Eclipse - in third_party/eclipse
gsutil = in third_party/goutil
Guava - in third_party/guava
hamcrest - in third_party/hamcrest
HttpLib2 - in samples/third_party/httpLib2
JSON - in third_party/json
JUnit - in third_party/junit
Oauth - in samples/third_party/oauth2client
Rhino - in third_party/rhino
weberknecht - in third_party/weberknecht

The libraries may have their own licenses; we recommend you read them, as their terms may differ from the terms below.

Copyright 2012, the Dart project authors. All rights reserved.
Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions are
met:

- * Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above
copyright notice, this list of conditions and the following
disclaimer in the documentation and/or other materials provided
with the distribution.
- * Neither the name of Google Inc. nor the names of its
contributors may be used to endorse or promote products derived
from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS
"AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT
LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR
A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT
OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE
OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

libffi - Copyright (c) 1996-2014 Anthony Green, Red Hat, Inc and others.

See source files for details.

Permission is hereby granted, free of charge, to any person obtaining
a copy of this software and associated documentation files (the
``Software"), to deal in the Software without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to
permit persons to whom the Software is furnished to do so, subject to
the following conditions:

The above copyright notice and this permission notice shall be
included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED ``AS IS'', WITHOUT WARRANTY OF ANY KIND,
EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The Apache 2 license (given in full in LICENSE.txt) applies to all code in this repository which is copyright

by JetBrains. The following sections of the repository contain third-party code, to which different licenses may apply:

Kotlin Compiler

The following modules contain third-party code and are incorporated into the Kotlin compiler and/or the Kotlin IntelliJ IDEA plugin:

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxStackSizeAndLocalsCalculator.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/inline/MaxLocalsCalculator.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/FastMethodAnalyzer.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/InstructionLivenessAnalyzer.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/ControlFlowGraph.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/fixStack/FastStackAnalyzer.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom

- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/temporaryVals/FastStoreLoadAnalyzer.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])

- Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: core/reflection/jvm/src/kotlin.reflect/jvm/internal/pcollections
 - License: MIT ([license/third_party/pcollections_LICENSE.txt][pcollections])
 - Origin: Derived from PCollections, A Persistent Java Collections Library (<https://pcollections.org/>)
- Path: eval4j/src/org/jetbrains/eval4j/interpreterLoop.kt
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: compiler/backend/src/org/jetbrains/kotlin/codegen/optimization/common/OptimizationBasicInterpreter.java
 - License: BSD ([license/third_party/asm_license.txt][asm])
 - Origin: Derived from ASM: a very small and fast Java bytecode manipulation framework, Copyright (c) 2000-2011 INRIA, France Telecom
- Path: js/js.ast
 - License: BSD ([license/third_party/dart_LICENSE.txt][dart])
 - Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,
- Path: js/js.inliner/src/org/jetbrains/kotlin/js/inline/FunctionInlineMutator.kt
 - License: BSD ([license/third_party/dart_LICENSE.txt][dart])
 - Origin: Originally part of the Dart compiler, (c) 2011 the Dart Project Authors,
- Path: js/js.parser/src/com/google
 - License: Netscape Public License 1.1 ([license/third_party/rhino_LICENSE.txt][rhino])
 - Origin: Originally part of GWT, (C) 2007-08 Google Inc., distributed under the Apache 2 license. The code is derived from Rhino, (C) 1997-1999 Netscape Communications Corporation, distributed under the Netscape Public License.
- Path: js/js.translator/qunit/qunit.js
 - License: MIT ([license/third_party/qunit_license.txt][qunit])
 - Origin: QUnit, Copyright (c) 2012 John Resig, Jrn Zaeferrer,
- Path: libraries/stdlib/src/kotlin/collections
 - License: Apache 2 ([license/third_party/gwt_license.txt][gwt])
 - Origin: Derived from GWT, (C) 2007-08 Google Inc.
- Path: libraries/stdlib/unsigned/src/kotlin/UnsignedUtils.kt
 - License: Apache 2 ([license/third_party/guava_license.txt][guava])
 - Origin: Derived from Guava's UnsignedLongs, (C) 2011 The Guava Authors
- Path: libraries/stdlib/jvm/src/kotlin/util/MathJVM.kt
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.
- Path: libraries/stdlib/js/src/kotlin/collections

- License: Apache 2 ([license/third_party/gwt_license.txt][gwt])
- Origin: Derived from GWT, (C) 2007-08 Google Inc.

- Path: libraries/stdlib/js-v1/src/js/long.js
 - License: Apache 2 ([license/third_party/closure-compiler_LICENSE.txt][closure-compiler])
 - Origin: Google Closure Library, Copyright 2009 The Closure Library Authors

- Path: libraries/stdlib/js-v1/src/js/polyfills.js
 - License: Boost Software License 1.0 ([license/third_party/boost_LICENSE.txt][boost])
 - Origin: Derived from boost special math functions, Copyright Eric Ford & Hubert Holin 2001.

- Path: libraries/stdlib/wasm/internal/kotlin/wasm/internal/Number2String.kt
 - License: Apache 2 ([third_party/assemblyscript_license.txt][assemblyscript])
 - Origin: Derived from assemblyscript standard library

- Path: plugins/lint/android-annotations
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: plugins/lint/lint-api
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: plugins/lint/lint-checks
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: plugins/lint/lint-idea
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: wasm/ir/src/org/jetbrains/kotlin/wasm/ir/convertors
 - License: MIT ([license/third_party/asmble_license.txt][asmble])
 - Origin: Copyright (C) 2018 Chad Retz

- Path: compiler/tests-common/tests/org/jetbrains/kotlin/codegen/ir/ComposeLikeGenerationExtension.kt
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Derived from JetPack Compose compiler plugin code, Copyright 2019 The Android Open Source Project

Kotlin Test Data

The following source code is used for testing the Kotlin compiler and/or plugin and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: third-party/annotations/android
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: third-party/annotations/com/android
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: third-party/annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.

- Path: third-party/annotations/androidx
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: third-party/annotations/edu/umd/cs/findbugs
 - License: LGPL 2.1 ([license/third_party/testdata/findbugs_license.txt][findbugs])
 - Origin: Bytecode Analysis Framework, Copyright (C) 2005 University of Maryland

- Path: third-party/java8-annotations/org/eclipse
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
 - Origin: Eclipse JDT, Copyright (c) 2011, 2013 Stephan Herrmann and others.

- Path: third-party/annotations/io/reactivex
 - License: Apache 2 ([license/third_party/testdata/rxjava_license.txt][rxjava])
 - Origin: RxJava, Copyright (c) 2016-present, RxJava Contributors

- Path: third-party/java8-annotations/org/jspecify
 - License: Apache 2 ([license/third_party/testdata/jspecify_license.txt][jspecify])
 - Origin: JSpecify, Copyright (C) 2020 The JSpecify Authors

- Path: third-party/java9-annotations/org/jspecify
 - License: Apache 2 ([license/third_party/testdata/jspecify_license.txt][jspecify])
 - Origin: JSpecify, Copyright (C) 2020 The JSpecify Authors

- Path: third-party/annotations/lombok
 - License: MIT ([license/third_party/testdata/lombok_license.txt][lombok])
 - Origin: Project Lombok, Copyright (C) 2009-2013 The Project Lombok Authors

- Path: idea/idea-android/tests/org/jetbrains/kotlin/android/AndroidTestBase.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lintQuickfix/requiresApi/RequiresApi.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/IntRange.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: idea/testData/android/lint/RequiresPermission.java
 - License: Apache 2 ([license/third_party/aosp_license.txt][aosp])
 - Origin: Copyright (C) 2011-15 The Android Open Source Project

- Path: libraries/tools/kotlin-gradle-plugin-integration-
tests/src/test/resources/testProject/allOpenSpring/src/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/AndroidDaggerProject
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/resources/testProject/kapt2
 - License: Apache 2 ([license/third_party/testdata/dagger_license.txt][dagger])
 - Origin: Dagger, Copyright (C) 2013 Square, Inc.

- Path: libraries/tools/kotlin-maven-plugin-test/src/it/test-allopen-
spring/src/main/java/org/springframework/stereotype/Component.java
 - License: Apache 2 ([license/third_party/testdata/spring_license.txt][spring])
 - Origin: Spring Framework, Copyright 2002-2007 the original author or authors.

Kotlin Tools and Libraries Tests

The following source code is used for testing the Kotlin tools and/or libraries and is not incorporated into any distributions of the tools or libraries:

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/src/test/kotlin/org/jetbrains/kotlin/gradle/PluginsDslIT.kt
 - License: Apache 2 ([license/third_party/gradle_license.txt][gradle])
 - Origin: Gradle, Copyright 2002-2017 Gradle, Inc.

- Path: libraries/tools/kotlin-gradle-plugin-integration-tests/target/test-
classes/testProject/noArgJpa/src/javax/persistence/Entity.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
and Eclipse Distribution License - v1.0
([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
 - Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

- Path: libraries/tools/kotlin-gradle-plugin-integration-
tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Entity.java
 - License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
and Eclipse Distribution License - v1.0
([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
 - Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

- Path: libraries/tools/kotlin-gradle-plugin-integration-
tests/src/test/resources/testProject/noArgJpa/src/javax/persistence/Embeddable.java

- License: Eclipse Public License v1.0 ([license/third_party/testdata/eclipse_license.txt][eclipse])
and Eclipse Distribution License - v1.0
([license/third_party/testdata/eclipse_distribution_license.txt][eclipse-distribution])
- Origin: javax.persistence, Copyright (c) 2008, 2017 Sun Microsystems, Oracle Corporation.

- Path: libraries/tools/kotlin-test-js-runner/karma-kotlin-reporter.js
- License: MIT ([license/third_party/karma_LICENSE.txt](third_party/karma_LICENSE.txt)
and [license/third_party/karma-teamcity-reporter_LICENSE.txt](third_party/karma-teamcity-reporter_LICENSE.txt))
- Origin: Copyright (C) 2011-2019 Google, Inc. and Copyright (C) 2011-2013 Vojta Jna and contributors.

- Path: libraries/tools/kotlin-test-js-runner/mocha-kotlin-reporter.js
- License: MIT ([license/third_party/mocha-teamcity-reporter_LICENSE.txt](third_party/mocha-teamcity-reporter_LICENSE.txt))
- Origin: Copyright (c) 2016 Jamie Sherriff

- Path: libraries/tools/kotlin-test-js-runner/src/utils.ts
- License: MIT ([license/third_party/teamcity-service-messages_LICENSE.txt](third_party/teamcity-service-messages_LICENSE.txt)
and [license/third_party/lodash_LICENSE.txt](third_party/lodash_LICENSE.txt))
- Origin: Copyright (c) 2013 Aaron Forsander and Copyright JS Foundation and other contributors
<<https://js.foundation/>>

- Path: libraries/tools/kotlin-test-js-runner/src/teamcity-format.js
- License: MIT ([license/third_party/mocha-teamcity-reporter_LICENSE.txt](third_party/mocha-teamcity-reporter_LICENSE.txt)
and [license/third_party/teamcity-service-messages_LICENSE.txt](third_party/teamcity-service-messages_LICENSE.txt))
- Origin: Copyright (c) 2016 Jamie Sherriff and Copyright (c) 2013 Aaron Forsander

Example Code

The following code is provided as examples and is not incorporated into any distributions of the compiler, libraries or plugin:

- Path: libraries/examples/browser-example/src/js/jquery.js
- License: MIT ([license/third_party/jquery_license.txt][jquery])
- Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

- Path: libraries/examples/browser-example-with-library/src/js/jquery.js
- License: MIT ([license/third_party/jquery_license.txt][jquery])
- Origin: jQuery JavaScript Library v1.6.2, Copyright 2011, John Resig

[aosp]: third_party/aosp_license.txt
 [asm]: third_party/asm_license.txt
 [boost]: third_party/boost_LICENSE.txt
 [closure-compiler]: third_party/closure-compiler_LICENSE.txt
 [dagger]: third_party/testdata/dagger_license.txt

[dart]: third_party/dart_LICENSE.txt
[eclipse]: third_party/testdata/eclipse_license.txt
[eclipse-distribution]: third_party/testdata/eclipse_distribution_license.txt
[findbugs]: third_party/testdata/findbugs_license.txt
[gradle]: third_party/gradle_license.txt
[guava]: third_party/guava_license.txt
[gwt]: third_party/gwt_license.txt
[jquery]: third_party/jquery_license.txt
[lombok]: third_party/testdata/lombok_license.txt
[pcollections]: third_party/pcollections_LICENSE.txt
[qunit]: third_party/qunit_license.txt
[rhino]: third_party/rhino_LICENSE.txt
[rxjava]: third_party/testdata/rxjava_license.txt
[spring]: third_party/testdata/spring_license.txt
[assemblyscript]: third_party/assemblyscript_license.txt
/*
* Copyright 2010-2021 JetBrains s.r.o. and Kotlin Programming Language contributors.
* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.
*/

GCC RUNTIME LIBRARY EXCEPTION

Version 3.1, 31 March 2009

Copyright (C) 2009 Free Software Foundation, Inc.

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This GCC Runtime Library Exception ("Exception") is an additional permission under section 7 of the GNU General Public License, version 3 ("GPLv3"). It applies to a given file (the "Runtime Library") that bears a notice placed by the copyright holder of the file stating that the file is governed by GPLv3 along with this Exception.

When you use GCC to compile a program, GCC may combine portions of certain GCC header files and runtime libraries with the compiled program. The purpose of this Exception is to allow compilation of non-GPL (including proprietary) programs to use, in this way, the header files and runtime libraries covered by this Exception.

0. Definitions.

A file is an "Independent Module" if it either requires the Runtime Library for execution after a Compilation Process, or makes use of an interface provided by the Runtime Library, but is not otherwise based on the Runtime Library.

"GCC" means a version of the GNU Compiler Collection, with or without

modifications, governed by version 3 (or a specified later version) of the GNU General Public License (GPL) with the option of using any subsequent versions published by the FSF.

"GPL-compatible Software" is software whose conditions of propagation, modification and use would permit combination with GCC in accord with the license of GCC.

"Target Code" refers to output from any compiler for a real or virtual target processor architecture, in executable form or suitable for input to an assembler, loader, linker and/or execution phase. Notwithstanding that, Target Code does not include data in any format that is used as a compiler intermediate representation, or used for producing a compiler intermediate representation.

The "Compilation Process" transforms code entirely represented in non-intermediate languages designed for human-written code, and/or in Java Virtual Machine byte code, into Target Code. Thus, for example, use of source code generators and preprocessors need not be considered part of the Compilation Process, since the Compilation Process can be understood as starting with the output of the generators or preprocessors.

A Compilation Process is "Eligible" if it is done using GCC, alone or with other GPL-compatible software, or if it is done without using any work based on GCC. For example, using non-GPL-compatible Software to optimize any GCC intermediate representations would not qualify as an Eligible Compilation Process.

1. Grant of Additional Permission.

You have permission to propagate a work of Target Code formed by combining the Runtime Library with Independent Modules, even if such propagation would otherwise violate the terms of GPLv3, provided that all Target Code was generated by Eligible Compilation Processes. You may then convey such a combination under terms of your choice, consistent with the licensing of the Independent Modules.

2. No Weakening of GCC Copyleft.

The availability of this Exception does not imply any general presumption that third-party software is unaffected by the copyleft requirements of the license of GCC.

Hopefully that text is self-explanatory. If it isn't, you need to speak to your lawyer, or the Free Software Foundation.
Copyright (c) 2008 Harold Cooper

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Eclipse Public License, Version 1.0 (EPL-1.0)

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

- a) in the case of the initial Contributor, the initial code and documentation distributed under this Agreement, and
- b) in the case of each subsequent Contributor:
 - i) changes to the Program, and
 - ii) additions to the Program;

where such changes and/or additions to the Program originate from and are distributed by that particular Contributor. A Contribution 'originates' from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include additions to the Program which: (i) are separate modules of software distributed in conjunction with the Program under their own license agreement, and (ii) are not derivative works of the Program.

"Contributor" means any person or entity that distributes the Program.

"Licensed Patents " mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement, including all Contributors.

2. GRANT OF RIGHTS

- a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Contribution of such Contributor, if any, and such derivative works, in source code and object code form.
- b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in source code and object code form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution.
No hardware per se is licensed hereunder.
- c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.
- d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

3. REQUIREMENTS

A Contributor may choose to distribute the Program in object code form under

its own license agreement, provided that:

- a) it complies with the terms and conditions of this Agreement; and
- b) its license agreement:
 - i) effectively disclaims on behalf of all Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;
 - ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;
 - iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and
 - iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

- a) it must be made available under this Agreement; and
- b) a copy of this Agreement must be included with each copy of the Program.

Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program

in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must:

- a) promptly notify the Commercial Contributor in writing of such claim,
- and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement , including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by

the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to distribute the Program (including its Contributions) under the new version. Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved.

This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation.

JSON

Copyright (c) 2002 JSON.org

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights

to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

The Software shall be used for Good, not Evil.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

/*

* Copyright 2000-2018 JetBrains s.r.o. and Kotlin Programming Language contributors.

* Use of this source code is governed by the Apache 2.0 license that can be found in the license/LICENSE.txt file.

*/

package tasks

```
import groovy.util.Node
import groovy.util.XmlParser
import org.gradle.api.DefaultTask
import org.gradle.api.file.RegularFileProperty
import org.gradle.api.provider.Property
import org.gradle.api.tasks.Input
import org.gradle.api.tasks.InputFile
import org.gradle.api.tasks.OutputFile
import org.gradle.api.tasks.TaskAction
import java.util.*

abstract class WriteCopyrightToFile : DefaultTask() {
    @InputFile
    val path = project.file("${project.rootDir}/.idea/copyright/apache.xml")

    @get:OutputFile
    abstract val outputFile: RegularFileProperty

    @get:Input
    val commented: Property<Boolean> = project.objects.property(Boolean::class.java).convention(true)

    @TaskAction
    fun write() {
        val file = outputFile.asFile.get()
        file.writeText(if (commented.get()) readCopyrightCommented() else readCopyright())
    }
}
```

```

}

private fun readCopyright(): String {
    assert(path.exists()) {
        "File $path with copyright not found"
    }

    val xmlParser = XmlParser()
    val node = xmlParser.parse(path)
    assert(node.attribute("name") == "CopyrightManager") {
        "Format changed occasionally?"
    }

    val copyrightBlock = node.children().filterIsInstance<Node>().single()
    val noticeNode = copyrightBlock.children().filterIsInstance<Node>().single { it.attribute("name") == "notice" }
    return noticeNode.attribute("value").toString().replace("&#36;today.year",
GregorianCalendar()[Calendar.YEAR].toString())
}

private fun readCopyrightCommented(): String {
    return "/*\n" + readCopyright().prependIndent(" * ") + "\n */"
}
}

```

Apache License
 Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

The MIT License

Copyright (C) 2011-2013 Vojta Jna and contributors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR

IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
MIT License

Copyright (c) 2018 Chad Retz

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

```
=====
== NOTICE file corresponding to the section 4 d of      ==
== the Apache License, Version 2.0,                    ==
== in this case for the Kotlin Compiler distribution.   ==
=====
```

Kotlin Compiler

Copyright 2010-2017 JetBrains s.r.o and respective authors and developers

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not

pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special,

incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2000-2020 JetBrains s.r.o.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/**

* Copyright 2010 Tim Down.

*

* Licensed under the Apache License, Version 2.0 (the "License");

- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Copyright (C) 2009-2021 The Project Lombok Authors.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright (C) 2011-2019 Google, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER

IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

zlib.h -- interface of the 'zlib' general purpose compression library
version 1.2.11, January 15th, 2017

Copyright (C) 1995-2017 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied warranty. In no event will the authors be held liable for any damages arising from the use of this software.

Permission is granted to anyone to use this software for any purpose, including commercial applications, and to alter it and redistribute it freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not claim that you wrote the original software. If you use this software in a product, an acknowledgment in the product documentation would be appreciated but is not required.
2. Altered source versions must be plainly marked as such, and must not be misrepresented as being the original software.
3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly Mark Adler
jloup@gzip.org madler@alumni.caltech.edu

SCALA LICENSE

Copyright (c) 2002-2012 EPFL, Lausanne, unless otherwise specified.
All rights reserved.

This software was developed by the Programming Methods Laboratory of the Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland.

Permission to use, copy, modify, and distribute this software in source or binary form for any purpose with or without fee is hereby granted, provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the EPFL nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

ASM: a very small and fast Java bytecode manipulation framework
Copyright (c) 2000-2005 INRIA, France Telecom
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The MIT License (MIT)

Copyright (c) 2016 Jamie Sherriff

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell

copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Licenses for included components:

Eclipse Public License 1.0

<https://opensource.org/licenses/EPL-1.0>

junit:junit

org.sonatype.aether:aether-api

org.sonatype.aether:aether-connector-wagon

org.sonatype.aether:aether-impl

org.sonatype.aether:aether-spi

org.sonatype.aether:aether-util

3-Clause BSD

<https://opensource.org/licenses/BSD-3-Clause>

com.google.code.findbugs:jsr305

org.hamcrest:hamcrest-core

BSD License

Copyright (c) 2000-2015 www.hamcrest.org

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Hamcrest nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY

EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT

SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED

TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR

BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

com.esotericsoftware.kryo:kryo

com.esotericsoftware.minlog:minlog

Copyright (c) 2008-2018, Nathan Sweet All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Esoteric Software nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

org.ow2.asm:asm

org.ow2.asm:asm-analysis

org.ow2.asm:asm-commons

org.ow2.asm:asm-tree

org.ow2.asm:asm-util

ASM: a very small and fast Java bytecode manipulation framework

Copyright (c) 2000-2011 INRIA, France Telecom

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holders nor the names of its

contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

MIT

com.googlecode.plist:dd-plist

dd-plist - An open source library to parse and generate property lists

Copyright (C) 2016 Daniel Dreibrodt

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

org.bouncycastle:bcpj-jdk15on

org.bouncycastle:bcprov-jdk15on

Copyright (c) 2000 - 2019 The Legion of the Bouncy Castle Inc. (<https://www.bouncycastle.org>)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

org.slf4j:jcl-over-slf4j
org.slf4j:jul-to-slf4j
org.slf4j:log4j-over-slf4j
org.slf4j:slf4j-api
Copyright (c) 2004-2017 QOS.ch
All rights reserved.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CDDL

<https://opensource.org/licenses/CDDL-1.0>

com.sun.xml.bind:jaxb-impl

LGPL 2.1

<https://www.gnu.org/licenses/old-licenses/lgpl-2.1.en.html>

org.samba.jcifs:jcifs

org.jetbrains.intellij.deps:trove4j

License for the GNU Trove library included by the Kotlin embeddable compiler

The source code for GNU Trove is licensed under the Lesser GNU Public License (LGPL).

Copyright (c) 2001, Eric D. Friedman All Rights Reserved. This library is free software; you can redistribute it and/or modify it under

the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or

(at your option) any later version. This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without

even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this program; if not, write to the Free Software

Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111-1307, USA.

Two classes (HashFunctions and PrimeFinder) included in Trove are licensed under the following terms:

Copyright (c) 1999 CERN - European Organization for Nuclear Research. Permission to use, copy, modify, distribute and sell this software

and its documentation for any purpose is hereby granted without fee, provided that the above copyright notice appear in all copies and

that both that copyright notice and this permission notice appear in supporting documentation. CERN makes no representations about the

suitability of this software for any purpose. It is provided "as is" without expressed or implied warranty.

The source code of modified GNU Trove library is available at

<https://github.com/JetBrains/intellij-deps-trove4j> (with trove4j_changes.txt describing the changes)

Eclipse Distribution License 1.0

<https://www.eclipse.org/org/documents/edl-v10.php>

org.eclipse.jgit:org.eclipse.jgit

BSD-style

com.jcraft:jsch

com.jcraft:jzlib

Copyright (c) 2000-2011 ymnk, JCraft, Inc. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The names of the authors may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL JCRAFT, INC. OR ANY CONTRIBUTORS TO THIS SOFTWARE BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Eclipse Public License 2.0

<https://www.eclipse.org/legal/epl-2.0/>

org.junit.platform:junit-platform-launcher

Mozilla Public License 2.0

<https://www.mozilla.org/en-US/MPL/2.0/>

org.mozilla:rhino

/*

* Copyright 2010-2020 JetBrains s.r.o. and Kotlin Programming Language contributors.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

must as a whole is licensed under the following standard MIT license:

Copyright 2005-2014 Rich Felker, et al.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Authors/contributors include:

Anthony G. Basile
Arvid Picciani
Bobby Bingham
Boris Brezillon
Chris Spiegel
Emil Renner Berthing
Hiltjo Posthuma
Isaac Dunham
Jens Gustedt
Jeremy Huntwork
John Spencer
Justin Cormack
Luca Barbato
Luka Perkov
Michael Forney
Nicholas J. Kain
orc
Pascal Cuoq

Pierre Carrier
Rich Felker
Richard Pennington
Solar Designer
Strake
Szabolcs Nagy
Timo Ters
Valentin Ochs
William Haddon

Portions of this software are derived from third-party works licensed under terms compatible with the above MIT license:

The TRE regular expression implementation (`src/regex/reg*` and `src/regex/tre*`) is Copyright 2001-2008 Ville Laurikari and licensed under a 2-clause BSD license (license text in the source files). The included version has been heavily modified by Rich Felker in 2012, in the interests of size, simplicity, and namespace cleanliness.

Much of the math library code (`src/math/*` and `src/complex/*`) is Copyright 1993,2004 Sun Microsystems or Copyright 2003-2011 David Schultz or Copyright 2003-2009 Steven G. Kargl or Copyright 2003-2009 Bruce D. Evans or Copyright 2008 Stephen L. Moshier and labelled as such in comments in the individual source files. All have been licensed under extremely permissive terms.

The ARM memcpy code (`src/string/armel/memcpy.s`) is Copyright 2008 The Android Open Source Project and is licensed under a two-clause BSD license. It was taken from Bionic libc, used on Android.

The implementation of DES for crypt (`src/misc/crypt_des.c`) is Copyright 1994 David Burren. It is licensed under a BSD license.

The implementation of blowfish crypt (`src/misc/crypt_blowfish.c`) was originally written by Solar Designer and placed into the public domain. The code also comes with a fallback permissive license for use in jurisdictions that may not recognize the public domain.

The smoothsort implementation (`src/stdlib/qsort.c`) is Copyright 2011 Valentin Ochs and is licensed under an MIT-style license.

The BSD PRNG implementation (`src/prng/random.c`) and XSI search API (`src/search/*.c`) functions are Copyright 2011 Szabolcs Nagy and licensed under following terms: "Permission to use, copy, modify, and/or distribute this code for any purpose with or without fee is hereby granted. There is no warranty."

The x86_64 port was written by Nicholas J. Kain. Several files (crt) were released into the public domain; others are licensed under the standard MIT license terms at the top of this file. See individual files for their copyright status.

The mips and microblaze ports were originally written by Richard Pennington for use in the elcc project. The original code was adapted by Rich Felker for build system and code conventions during upstream integration. It is licensed under the standard MIT terms.

The powerpc port was also originally written by Richard Pennington, and later supplemented and integrated by John Spencer. It is licensed under the standard MIT terms.

All other files which have no copyright comments are original works produced specifically for use as part of this library, written either by Rich Felker, the main author of the library, or by one or more contributors listed above. Details on authorship of individual files can be found in the git version control history of the project. The omission of copyright and license comments in each file is in the interest of source tree size.

All public header files (include/* and arch/*/bits/*) should be treated as Public Domain as they intentionally contain no content which can be covered by copyright. Some source modules may fall in this category as well. If you believe that a file is so trivial that it should be in the Public Domain, please contact the authors and request an explicit statement releasing it from copyright.

The following files are trivial, believed not to be copyrightable in the first place, and hereby explicitly released to the Public Domain:

All public headers: include/*, arch/*/bits/*

Startup files: crt/*

UNLESS SPECIFIED OTHERWISE IN THE INDIVIDUAL SOURCE FILES INCLUDED WITH THIS PACKAGE, they may freely be used, modified and/or redistributed for any purpose.

Portions, Copyright 1991-2005 Unicode, Inc. The following applies to Unicode.

COPYRIGHT AND PERMISSION NOTICE

Copyright 1991-2005 Unicode, Inc. All rights reserved. Distributed under the Terms of Use in <http://www.unicode.org/copyright.html>. Permission is hereby granted, free of charge, to any person obtaining a copy of the Unicode data files and any associated documentation (the "Data Files") or Unicode software and any associated documentation (the "Software") to deal in the Data Files or Software without restriction, including without

limitation the rights to use, copy, modify, merge, publish, distribute, and/or sell copies of the Data Files or Software, and to permit persons to whom the Data Files or Software are furnished to do so, provided that (a) the above copyright notice(s) and this permission notice appear with all copies of the Data Files or Software, (b) both the above copyright notice(s) and this permission notice appear in associated documentation, and (c) there is clear notice in each modified Data File or in the Software as well as in the documentation associated with the Data File(s) or Software that the data or software has been modified.

THE DATA FILES AND SOFTWARE ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT OF THIRD PARTY RIGHTS. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR HOLDERS INCLUDED IN THIS NOTICE BE LIABLE FOR ANY CLAIM, OR ANY SPECIAL INDIRECT OR CONSEQUENTIAL DAMAGES, OR ANY DAMAGES WHATSOEVER RESULTING FROM LOSS OF USE, DATA OR PROFITS, WHETHER IN AN ACTION OF CONTRACT, NEGLIGENCE OR OTHER TORTIOUS ACTION, ARISING OUT OF OR IN CONNECTION WITH THE USE OR PERFORMANCE OF THE DATA FILES OR SOFTWARE.

Except as contained in this notice, the name of a copyright holder shall not be used in advertising or otherwise to promote the sale, use or other dealings in these Data Files or Software without prior written authorization of the copyright holder.

2. Additional terms from the Database:

Copyright 1995-1999 Unicode, Inc. All Rights reserved.

Disclaimer

The Unicode Character Database is provided as is by Unicode, Inc. No claims are made as to fitness for any particular purpose. No warranties of any kind are expressed or implied. The recipient agrees to determine applicability of information provided. If this file has been purchased on magnetic or optical media from Unicode, Inc., the sole remedy for any claim will be exchange of defective media within 90 days of receipt. This disclaimer is applicable for all other data files accompanying the Unicode Character Database, some of which have been compiled by the Unicode Consortium, and some of which have been supplied by other sources.

Limitations on Rights to Redistribute This Data

Recipient is granted the right to make copies in any form for internal distribution and to freely use the information supplied in the creation of products supporting the Unicode™ Standard. The files in the Unicode Character Database can be redistributed to third parties or other organizations (whether for profit or not) as long as this notice and the disclaimer

notice are retained. Information can be extracted from these files and used in documentation or programs, as long as there is an accompanying notice indicating the source.

Google Dart Js backend was removed - <https://code.google.com/p/dart/source/detail?r=4771>

According to <http://www.apache.org/legal/3party.html> we can include "Google Dart Js backend" in source form, because code license is "New BSD License" (Authorized License).

This part of code will be removed when kotlin will be rewritten on kotlin.
Copyright JS Foundation and other contributors, <https://js.foundation/>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/jquery/jquery>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Boost Software License - Version 1.0 - August 17th, 2003

Permission is hereby granted, free of charge, to any person or organization obtaining a copy of the software and accompanying documentation covered by this license (the "Software") to use, reproduce, display, distribute, execute, and transmit the Software, and to prepare derivative works of the Software, and to permit third-parties to whom the Software is furnished to do so, all subject to the following:

The copyright notices in the Software and this entire statement, including the above license grant, this restriction and the following disclaimer, must be included in all copies of the Software, in whole or in part, and all derivative works of the Software, unless such copies or derivative works are solely in the form of machine-executable object code generated by a source language processor.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR ANYONE DISTRIBUTING THE SOFTWARE BE LIABLE FOR ANY DAMAGES OR OTHER LIABILITY, WHETHER IN CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The files in this package are taken from mozilla's Rhino project.
See <http://www.mozilla.org/rhino/>

The files modified from Rhino 1.5R3
(<ftp://ftp.mozilla.org/pub/js/rhino15R3.zip>).

--

The contents of this package are subject to the Netscape Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/NPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is Rhino code, released
May 6, 1999.

The Initial Developer of the Original Code is Netscape Communications Corporation. Portions created by Netscape are Copyright (C) 1997-2000 Netscape Communications Corporation. All Rights Reserved.

Alternatively, the contents of this file may be used under the terms of the GNU Public License (the "GPL"), in which case the provisions of the GPL are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the GPL and not to allow others to use your version of this file under the NPL, indicate your decision by deleting the provisions above and replace them with the notice

and other provisions required by the GPL. If you do not delete the provisions above, a recipient may use your version of this file under either the NPL or the GPL.
<http://fastutil.di.unimi.it/>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

The majority of Rhino is MPL 1.1 / GPL 2.0 dual licensed:

The Mozilla Public License (<http://www.mozilla.org/MPL/MPL-1.1.txt>):

=====

MOZILLA PUBLIC LICENSE

Version 1.1

1. Definitions.

1.0.1. "Commercial Use" means distribution or otherwise making the
Covered Code available to a third party.

1.1. "Contributor" means each entity that creates or contributes to
the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original
Code, prior Modifications used by a Contributor, and the Modifications
made by that particular Contributor.

1.3. "Covered Code" means the Original Code or Modifications or the
combination of the Original Code and Modifications, in each case
including portions thereof.

1.4. "Electronic Distribution Mechanism" means a mechanism generally
accepted in the software development community for the electronic
transfer of data.

1.5. "Executable" means Covered Code in any form other than Source
Code.

1.6. "Initial Developer" means the individual or entity identified
as the Initial Developer in the Source Code notice required by Exhibit

A.

1.7. "Larger Work" means a work which combines Covered Code or portions thereof with code not governed by the terms of this License.

1.8. "License" means this document.

1.8.1. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means any addition to or deletion from the substance or structure of either the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

A. Any addition to or deletion from the contents of a file containing Original Code or previous Modifications.

B. Any new file that contains any part of the Original Code or previous Modifications.

1.10. "Original Code" means Source Code of computer software code which is described in the Source Code notice required by Exhibit A as Original Code, and which, at the time of its release under this License is not already Covered Code governed by this License.

1.10.1. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.11. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice. The Source Code can be in a compressed or archival form, provided the appropriate decompression or de-archiving software is widely available for no charge.

1.12. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License or a future version of this License issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent

(50%) of the outstanding shares or beneficial ownership of such entity.

2. Source Code License.

2.1. The Initial Developer Grant.

The Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license, subject to third party intellectual property claims:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer to use, reproduce, modify, display, perform, sublicense and distribute the Original Code (or portions thereof) with or without Modifications, and/or as part of a Larger Work; and

(b) under Patents Claims infringed by the making, using or selling of Original Code, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Code (or portions thereof).

(c) the licenses granted in this Section 2.1(a) and (b) are effective on the date Initial Developer first distributes Original Code under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: 1) for code that You delete from the Original Code; 2) separate from the Original Code; or 3) for infringements caused by: i) the modification of the Original Code or ii) the combination of the Original Code with other software or devices.

2.2. Contributor Grant.

Subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor, to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof) either on an unmodified basis, with other Modifications, as Covered Code and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: 1) Modifications made by that Contributor (or portions thereof); and 2) the combination of Modifications made by that Contributor with its Contributor

Version (or portions of such combination).

(c) the licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first makes Commercial Use of the Covered Code.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: 1) for any code that Contributor has deleted from the Contributor Version; 2) separate from the Contributor Version; 3) for infringements caused by: i) third party modifications of Contributor Version or ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or 4) under Patent Claims infringed by Covered Code in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Application of License.

The Modifications which You create or to which You contribute are governed by the terms of this License, including without limitation Section 2.2. The Source Code version of Covered Code may be distributed only under the terms of this License or a future version of this License released under Section 6.1, and You must include a copy of this License with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this License or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.5.

3.2. Availability of Source Code.

Any Modification which You create or to which You contribute must be made available in Source Code form under the terms of this License either on the same media as an Executable version or via an accepted Electronic Distribution Mechanism to anyone to whom you made an Executable version available; and if made available via Electronic Distribution Mechanism, must remain available for at least twelve (12) months after the date it initially became available, or at least six (6) months after a subsequent version of that particular Modification has been made available to such recipients. You are responsible for ensuring that the Source Code version remains available even if the Electronic Distribution Mechanism is maintained by a third party.

3.3. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that

the Modification is derived, directly or indirectly, from Original Code provided by the Initial Developer and including the name of the Initial Developer in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

3.4. Intellectual Property Matters

(a) Third Party Claims.

If Contributor has knowledge that a license under a third party's intellectual property rights is required to exercise the rights granted by such Contributor under Sections 2.1 or 2.2, Contributor must include a text file with the Source Code distribution titled "LEGAL" which describes the claim and the party making the claim in sufficient detail that a recipient will know whom to contact. If Contributor obtains such knowledge after the Modification is made available as described in Section 3.2, Contributor shall promptly modify the LEGAL file in all copies Contributor makes available thereafter and shall take other steps (such as notifying appropriate mailing lists or newsgroups) reasonably calculated to inform those who received the Covered Code that new knowledge has been obtained.

(b) Contributor APIs.

If Contributor's Modifications include an application programming interface and Contributor has knowledge of patent licenses which are reasonably necessary to implement that API, Contributor must also include this information in the LEGAL file.

(c) Representations.

Contributor represents that, except as disclosed pursuant to Section 3.4(a) above, Contributor believes that Contributor's Modifications are Contributor's original creation(s) and/or Contributor has sufficient rights to grant the rights conveyed by this License.

3.5. Required Notices.

You must duplicate the notice in Exhibit A in each file of the Source Code. If it is not possible to put such notice in a particular Source Code file due to its structure, then You must include such notice in a location (such as a relevant directory) where a user would be likely to look for such a notice. If You created one or more Modification(s) You may add your name as a Contributor to the notice described in Exhibit A. You must also duplicate this License in any documentation for the Source Code where You describe recipients' rights or ownership rights relating to Covered Code. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Code. However, You may do so only on Your own behalf, and not on behalf of the Initial

Developer or any Contributor. You must make it absolutely clear than any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.6. Distribution of Executable Versions.

You may distribute Covered Code in Executable form only if the requirements of Section 3.1-3.5 have been met for that Covered Code, and if You include a notice stating that the Source Code version of the Covered Code is available under the terms of this License, including a description of how and where You have fulfilled the obligations of Section 3.2. The notice must be conspicuously included in any notice in an Executable version, related documentation or collateral in which You describe recipients' rights relating to the Covered Code. You may distribute the Executable version of Covered Code or ownership rights under a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable version does not attempt to limit or alter the recipient's rights in the Source Code version from the rights set forth in this License. If You distribute the Executable version under a different license You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or any Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.7. Larger Works.

You may create a Larger Work by combining Covered Code with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Code.

4. Inability to Comply Due to Statute or Regulation.

If it is impossible for You to comply with any of the terms of this License with respect to some or all of the Covered Code due to statute, judicial order, or regulation then You must: (a) comply with the terms of this License to the maximum extent possible; and (b) describe the limitations and the code they affect. Such description must be included in the LEGAL file described in Section 3.4 and must be included with all distributions of the Source Code. Except to the extent prohibited by statute or regulation, such description must be sufficiently detailed for a recipient of ordinary skill to be able to understand it.

5. Application of this License.

This License applies to code to which the Initial Developer has attached the notice in Exhibit A and to related Covered Code.

6. Versions of the License.

6.1. New Versions.

Netscape Communications Corporation ("Netscape") may publish revised and/or new versions of the License from time to time. Each version will be given a distinguishing version number.

6.2. Effect of New Versions.

Once Covered Code has been published under a particular version of the License, You may always continue to use it under the terms of that version. You may also choose to use such Covered Code under the terms of any subsequent version of the License published by Netscape. No one other than Netscape has the right to modify the terms applicable to Covered Code created under this License.

6.3. Derivative Works.

If You create or use a modified version of this License (which you may only do in order to apply it to code which is not already Covered Code governed by this License), You must (a) rename Your license so that the phrases "Mozilla", "MOZILLAPL", "MOZPL", "Netscape", "MPL", "NPL" or any confusingly similar phrase do not appear in your license (except to note that your license differs from this License) and (b) otherwise make it clear that Your version of the license contains terms which differ from the Mozilla Public License and Netscape Public License. (Filling in the name of the Initial Developer, Original Code or Contributor in the notice described in Exhibit A shall not of themselves be deemed to be modifications of this License.)

7. DISCLAIMER OF WARRANTY.

COVERED CODE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED CODE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED CODE IS WITH YOU. SHOULD ANY COVERED CODE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED CODE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

8. TERMINATION.

8.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. All sublicenses to the Covered Code which are properly granted shall survive any termination of this License. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

8.2. If You initiate litigation by asserting a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You file such action is referred to as "Participant") alleging that:

(a) such Participant's Contributor Version directly or indirectly infringes any patent, then any and all rights granted by such Participant to You under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively, unless if within 60 days after receipt of notice You either: (i) agree in writing to pay Participant a mutually agreeable reasonable royalty for Your past and future use of Modifications made by such Participant, or (ii) withdraw Your litigation claim with respect to the Contributor Version against such Participant. If within 60 days of notice, a reasonable royalty and payment arrangement are not mutually agreed upon in writing by the parties or the litigation claim is not withdrawn, the rights granted by Participant to You under Sections 2.1 and/or 2.2 automatically terminate at the expiration of the 60 day notice period specified above.

(b) any software, hardware, or device, other than such Participant's Contributor Version, directly or indirectly infringes any patent, then any rights granted to You by such Participant under Sections 2.1(b) and 2.2(b) are revoked effective as of the date You first made, used, sold, distributed, or had made, Modifications made by that Participant.

8.3. If You assert a patent infringement claim against Participant alleging that such Participant's Contributor Version directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

8.4. In the event of termination under Sections 8.1 or 8.2 above, all end user license agreements (excluding distributors and resellers)

which have been validly granted by You or any distributor hereunder prior to termination shall survive termination.

9. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED CODE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

10. U.S. GOVERNMENT END USERS.

The Covered Code is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" and "commercial computer software documentation," as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Code with only those rights set forth herein.

11. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by California law provisions (except to the extent applicable law, if any, provides otherwise), excluding its conflict-of-law provisions. With respect to disputes in which at least one party is a citizen of, or an entity chartered or registered to do business in the United States of America, any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California, with venue lying in Santa Clara County, California, with the losing party responsible for costs, including without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract

shall be construed against the drafter shall not apply to this License.

12. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

13. MULTIPLE-LICENSED CODE.

Initial Developer may designate portions of the Covered Code as "Multiple-Licensed". "Multiple-Licensed" means that the Initial Developer permits you to utilize portions of the Covered Code under Your choice of the NPL or the alternative licenses, if any, specified by the Initial Developer in the file described in Exhibit A.

EXHIBIT A -Mozilla Public License.

``The contents of this file are subject to the Mozilla Public License Version 1.1 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at <http://www.mozilla.org/MPL/>

Software distributed under the License is distributed on an "AS IS" basis, WITHOUT WARRANTY OF ANY KIND, either express or implied. See the License for the specific language governing rights and limitations under the License.

The Original Code is _____.

The Initial Developer of the Original Code is _____.

Portions created by _____ are Copyright (C) _____
_____. All Rights Reserved.

Contributor(s): _____.

Alternatively, the contents of this file may be used under the terms of the _____ license (the "[_____] License"), in which case the provisions of [_____] License are applicable instead of those above. If you wish to allow use of your version of this file only under the terms of the [_____] License and not to allow others to use your version of this file under the MPL, indicate your decision by deleting the provisions above and replace them with the notice and other provisions required by the [_____] License. If you do not delete

the provisions above, a recipient may use your version of this file under either the MPL or the [____] License."

[NOTE: The text of this Exhibit A may differ slightly from the text of the notices in the Source Code files of the Original Code. You should use the text of this Exhibit A rather than the text found in the Original Code Source Code for Your Modifications.]

GNU GENERAL PUBLIC LICENSE

Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.,
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Lesser General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

GNU GENERAL PUBLIC LICENSE

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,

under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are

prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the

original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the program's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

```
This program is free software; you can redistribute it and/or modify  
it under the terms of the GNU General Public License as published by  
the Free Software Foundation; either version 2 of the License, or  
(at your option) any later version.
```

```
This program is distributed in the hope that it will be useful,  
but WITHOUT ANY WARRANTY; without even the implied warranty of  
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the  
GNU General Public License for more details.
```

```
You should have received a copy of the GNU General Public License along  
with this program; if not, write to the Free Software Foundation, Inc.,  
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA.
```

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author  
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'.  
This is free software, and you are welcome to redistribute it  
under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if

necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

<signature of Ty Coon>, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with the
library. If this is what you want to do, use the GNU Lesser General
Public License instead of this License.

=====
Additionally, some files (currently the contents of
toolsrc/org/mozilla/javascript/tools/debugger/treetable/) are available
only under the following license:

- =====
* Copyright 1997, 1998 Sun Microsystems, Inc. All Rights Reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* - Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* - Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
*
* - Neither the name of Sun Microsystems nor the names of its
* contributors may be used to endorse or promote products derived
* from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS
* IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO,
* THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
* PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR
* CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL,
* EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO,
* PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR
* PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF
* LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING
* NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS
* SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2013 Aaron Forsander

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that

you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the

entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data

prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no

charge to all third parties under the terms of this License.

d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for

that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6,

whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the library's name and a brief idea of what it does.>
Copyright (C) <year> <name of author>

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the library, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!
The MIT License

Copyright (c) 2003, Kohsuke Kawaguchi

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

The MIT License

Copyright JS Foundation and other contributors <<https://js.foundation/>>

Based on Underscore.js, copyright Jeremy Ashkenas,
DocumentCloud and Investigative Reporters & Editors <<http://underscorejs.org/>>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/lodash/lodash>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION

WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

Copyright and related rights for sample code are waived via CC0. Sample code is defined as all source code displayed within the prose of the documentation.

CC0: <http://creativecommons.org/publicdomain/zero/1.0/>

====

Files located in the `node_modules` and `vendor` directories are externally maintained libraries used by this software which have their own licenses; we recommend you read them, as their terms may differ from the terms above.

Copyright JS Foundation and other contributors, <https://js.foundation>

This software consists of voluntary contributions made by many individuals. For exact contribution history, see the revision history available at <https://github.com/qunitjs/qunit>

The following license applies to all parts of this software except as documented below:

====

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

====

GNU LESSER GENERAL PUBLIC LICENSE

Version 2.1, February 1999

Copyright (C) 1991, 1999 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

[This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public Licenses are intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users.

This license, the Lesser General Public License, applies to some specially designated software packages--typically libraries--of the Free Software Foundation and other authors who decide to use it. You can use it too, but we suggest you first think carefully about whether this license or the ordinary General Public License is the better strategy to use in any particular case, based on the explanations below.

When we speak of free software, we are referring to freedom of use, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish); that you receive source code or can get it if you want it; that you can change the software and use pieces of it in new free programs; and that you are informed that you can do these things.

To protect your rights, we need to make restrictions that forbid distributors to deny you these rights or to ask you to surrender these rights. These restrictions translate to certain responsibilities for you if you distribute copies of the library or if you modify it.

For example, if you distribute copies of the library, whether gratis or for a fee, you must give the recipients all the rights that we gave you. You must make sure that they, too, receive or can get the source code. If you link other code with the library, you must provide complete object files to the recipients, so that they can relink them with the library after making changes to the library and recompiling it. And you must show them these terms so they know their rights.

We protect your rights with a two-step method: (1) we copyright the

library, and (2) we offer you this license, which gives you legal permission to copy, distribute and/or modify the library.

To protect each distributor, we want to make it very clear that there is no warranty for the free library. Also, if the library is modified by someone else and passed on, the recipients should know that what they have is not the original version, so that the original author's reputation will not be affected by problems that might be introduced by others.

Finally, software patents pose a constant threat to the existence of any free program. We wish to make sure that a company cannot effectively restrict the users of a free program by obtaining a restrictive license from a patent holder. Therefore, we insist that any patent license obtained for a version of the library must be consistent with the full freedom of use specified in this license.

Most GNU software, including some libraries, is covered by the ordinary GNU General Public License. This license, the GNU Lesser General Public License, applies to certain designated libraries, and is quite different from the ordinary General Public License. We use this license for certain libraries in order to permit linking those libraries into non-free programs.

When a program is linked with a library, whether statically or using a shared library, the combination of the two is legally speaking a combined work, a derivative of the original library. The ordinary General Public License therefore permits such linking only if the entire combination fits its criteria of freedom. The Lesser General Public License permits more lax criteria for linking other code with the library.

We call this license the "Lesser" General Public License because it does Less to protect the user's freedom than the ordinary General Public License. It also provides other free software developers Less of an advantage over competing non-free programs. These disadvantages are the reason we use the ordinary General Public License for many libraries. However, the Lesser license provides advantages in certain special circumstances.

For example, on rare occasions, there may be a special need to encourage the widest possible use of a certain library, so that it becomes a de-facto standard. To achieve this, non-free programs must be allowed to use the library. A more frequent case is that a free library does the same job as widely used non-free libraries. In this case, there is little to gain by limiting the free library to free software only, so we use the Lesser General Public License.

In other cases, permission to use a particular library in non-free programs enables a greater number of people to use a large body of free software. For example, permission to use the GNU C Library in non-free programs enables many more people to use the whole GNU operating system, as well as its variant, the GNU/Linux operating system.

Although the Lesser General Public License is Less protective of the users' freedom, it does ensure that the user of a program that is linked with the Library has the freedom and the wherewithal to run that program using a modified version of the Library.

The precise terms and conditions for copying, distribution and modification follow. Pay close attention to the difference between a "work based on the library" and a "work that uses the library". The former contains code derived from the library, whereas the latter must be combined with the library in order to run.

GNU LESSER GENERAL PUBLIC LICENSE TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License Agreement applies to any software library or other program which contains a notice placed by the copyright holder or other authorized party saying it may be distributed under the terms of this Lesser General Public License (also called "this License"). Each licensee is addressed as "you".

A "library" means a collection of software functions and/or data prepared so as to be conveniently linked with application programs (which use some of those functions and data) to form executables.

The "Library", below, refers to any such software library or work which has been distributed under these terms. A "work based on the Library" means either the Library or any derivative work under copyright law: that is to say, a work containing the Library or a portion of it, either verbatim or with modifications and/or translated straightforwardly into another language. (Hereinafter, translation is included without limitation in the term "modification".)

"Source code" for a work means the preferred form of the work for making modifications to it. For a library, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the library.

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running a program using the Library is not restricted, and output from

such a program is covered only if its contents constitute a work based on the Library (independent of the use of the Library in a tool for writing it). Whether that is true depends on what the Library does and what the program that uses the Library does.

1. You may copy and distribute verbatim copies of the Library's complete source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and distribute a copy of this License along with the Library.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Library or any portion of it, thus forming a work based on the Library, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) The modified work must itself be a software library.
- b) You must cause the files modified to carry prominent notices stating that you changed the files and the date of any change.
- c) You must cause the whole of the work to be licensed at no charge to all third parties under the terms of this License.
- d) If a facility in the modified Library refers to a function or a table of data to be supplied by an application program that uses the facility, other than as an argument passed when the facility is invoked, then you must make a good faith effort to ensure that, in the event an application does not supply such function or table, the facility still operates, and performs whatever part of its purpose remains meaningful.

(For example, a function in a library to compute square roots has a purpose that is entirely well-defined independent of the application. Therefore, Subsection 2d requires that any application-supplied function or table used by this function must be optional: if the application does not supply it, the square root function must still compute square roots.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Library, and can be reasonably considered independent and separate works in

themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Library, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Library.

In addition, mere aggregation of another work not based on the Library with the Library (or with a work based on the Library) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may opt to apply the terms of the ordinary GNU General Public License instead of this License to a given copy of the Library. To do this, you must alter all the notices that refer to this License, so that they refer to the ordinary GNU General Public License, version 2, instead of to this License. (If a newer version than version 2 of the ordinary GNU General Public License has appeared, then you can specify that version instead if you wish.) Do not make any other change in these notices.

Once this change is made in a given copy, it is irreversible for that copy, so the ordinary GNU General Public License applies to all subsequent copies and derivative works made from that copy.

This option is useful when you wish to copy part of the code of the Library into a program that is not a library.

4. You may copy and distribute the Library (or a portion or derivative of it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange.

If distribution of object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place satisfies the requirement to distribute the source code, even though third parties are not compelled to copy the source along with the object code.

5. A program that contains no derivative of any portion of the

Library, but is designed to work with the Library by being compiled or linked with it, is called a "work that uses the Library". Such a work, in isolation, is not a derivative work of the Library, and therefore falls outside the scope of this License.

However, linking a "work that uses the Library" with the Library creates an executable that is a derivative of the Library (because it contains portions of the Library), rather than a "work that uses the library". The executable is therefore covered by this License. Section 6 states terms for distribution of such executables.

When a "work that uses the Library" uses material from a header file that is part of the Library, the object code for the work may be a derivative work of the Library even though the source code is not. Whether this is true is especially significant if the work can be linked without the Library, or if the work is itself a library. The threshold for this to be true is not precisely defined by law.

If such an object file uses only numerical parameters, data structure layouts and accessors, and small macros and small inline functions (ten lines or less in length), then the use of the object file is unrestricted, regardless of whether it is legally a derivative work. (Executables containing this object code plus portions of the Library will still fall under Section 6.)

Otherwise, if the work is a derivative of the Library, you may distribute the object code for the work under the terms of Section 6. Any executables containing that work also fall under Section 6, whether or not they are linked directly with the Library itself.

6. As an exception to the Sections above, you may also combine or link a "work that uses the Library" with the Library to produce a work containing portions of the Library, and distribute that work under terms of your choice, provided that the terms permit modification of the work for the customer's own use and reverse engineering for debugging such modifications.

You must give prominent notice with each copy of the work that the Library is used in it and that the Library and its use are covered by this License. You must supply a copy of this License. If the work during execution displays copyright notices, you must include the copyright notice for the Library among them, as well as a reference directing the user to the copy of this License. Also, you must do one of these things:

- a) Accompany the work with the complete corresponding machine-readable source code for the Library including whatever changes were used in the work (which must be distributed under

Sections 1 and 2 above); and, if the work is an executable linked with the Library, with the complete machine-readable "work that uses the Library", as object code and/or source code, so that the user can modify the Library and then relink to produce a modified executable containing the modified Library. (It is understood that the user who changes the contents of definitions files in the Library will not necessarily be able to recompile the application to use the modified definitions.)

b) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (1) uses at run time a copy of the library already present on the user's computer system, rather than copying library functions into the executable, and (2) will operate properly with a modified version of the library, if the user installs one, as long as the modified version is interface-compatible with the version that the work was made with.

c) Accompany the work with a written offer, valid for at least three years, to give the same user the materials specified in Subsection 6a, above, for a charge no more than the cost of performing this distribution.

d) If distribution of the work is made by offering access to copy from a designated place, offer equivalent access to copy the above specified materials from the same place.

e) Verify that the user has already received a copy of these materials or that you have already sent this user a copy.

For an executable, the required form of the "work that uses the Library" must include any data and utility programs needed for reproducing the executable from it. However, as a special exception, the materials to be distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

It may happen that this requirement contradicts the license restrictions of other proprietary libraries that do not normally accompany the operating system. Such a contradiction means you cannot use both them and the Library together in an executable that you distribute.

7. You may place library facilities that are a work based on the Library side-by-side in a single library together with other library facilities not covered by this License, and distribute such a combined library, provided that the separate distribution of the work based on

the Library and of the other library facilities is otherwise permitted, and provided that you do these two things:

- a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities. This must be distributed under the terms of the Sections above.
- b) Give prominent notice with the combined library of the fact that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

8. You may not copy, modify, sublicense, link with, or distribute the Library except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, link with, or distribute the Library is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

9. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Library or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Library (or any work based on the Library), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Library or works based on it.

10. Each time you redistribute the Library (or any work based on the Library), the recipient automatically receives a license from the original licensor to copy, distribute, link with or modify the Library subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties with this License.

11. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Library at all. For example, if a patent license would not permit royalty-free redistribution of the Library by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to

refrain entirely from distribution of the Library.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply, and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

12. If the distribution and/or use of the Library is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Library under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

13. The Free Software Foundation may publish revised and/or new versions of the Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Library does not specify a license version number, you may choose any version ever published by the Free Software Foundation.

14. If you wish to incorporate parts of the Library into other free programs whose distribution conditions are incompatible with these, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing

and reuse of software generally.

NO WARRANTY

15. BECAUSE THE LIBRARY IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE LIBRARY, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE LIBRARY "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE LIBRARY IS WITH YOU. SHOULD THE LIBRARY PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE LIBRARY AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE LIBRARY (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE LIBRARY TO OPERATE WITH ANY OTHER SOFTWARE), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Libraries

If you develop a new library, and you want it to be of the greatest possible use to the public, we recommend making it free software that everyone can redistribute and change. You can do so by permitting redistribution under these terms (or, alternatively, under the terms of the ordinary General Public License).

To apply these terms, attach the following notices to the library. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<one line to give the library's name and a brief idea of what it does.>  
Copyright (C) <year> <name of author>
```

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public
License along with this library; if not, write to the Free Software
Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

You should also get your employer (if you work as a programmer) or your
school, if any, to sign a "copyright disclaimer" for the library, if
necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the
library `Frob' (a library for tweaking knobs) written by James Random Hacker.

<signature of Ty Coon>, 1 April 1990
Ty Coon, President of Vice

That's all there is to it!

1.317 jackson-module:-guice 2.13.2

1.317.1 Available under license :

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.
It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has
been in development since 2007.

It is currently developed by a community of developers, as well as supported
commercially by FasterXML.com.

Licensing

Jackson core and extension components may licensed under different licenses.
To find the details that apply to this artifact see the accompanying LICENSE file.
For more information, including possible other licensing options, contact
FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included
in some artifacts (usually source distributions); but is always available
from the source code management (SCM) system project uses.

This copy of Jackson JSON processor `jackson-module-guice` module is licensed under the Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.318 apiguardian-apiguardian-api 1.1.2

1.318.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.319 slf4j-api-module 1.7.25

1.320 automation 1.11-8

1.320.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * dk.brics.automaton
 *
 * Copyright (c) 2001-2011 Anders Moeller
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
```

* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):

* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/BasicAutomata.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/TransitionComparator.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/DatatypesAutomatonProvider.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Automaton.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Datatypes.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/SpecialOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/BasicOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/StatePair.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/MinimizationOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/State.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/ShuffleOperations.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/RunAutomaton.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/RegExp.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/AutomatonProvider.java
* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-jar/dk/brics/automaton/Transition.java

No license file was found, but licenses were detected in source scan.

/*
* dk.brics.automaton - AutomatonMatcher
*
* Copyright (c) 2008-2011 John Gibson

* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):

* /opt/cola/permits/1162863214_1620924489.9/0/automaton-1-11-8-sources-6-
jar/dk/brics/automaton/AutomatonMatcher.java

1.321 ktor-server-netty 1.2.6

1.321.1 Available under license :

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all

other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and

subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed

as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the

Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.322 antlr 3.5.2

1.322.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * [The "BSD license"]
 * Copyright (c) 2007-2008 Johannes Luber
 * Copyright (c) 2005-2007 Kunle Odutola
 * Copyright (c) 2011 Sam Harwell
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

@outputFile.imports() ::= <<
<@super.imports()>

<if(!TREE_PARSER)>
<! tree parser would already have imported !>
using Antlr.Runtime.Tree;
using RewriteRuleITokenStream = Antlr.Runtime.Tree.RewriteRuleTokenStream;
<endif>
>>

@genericParser.members() ::= <<
<@super.members()>
<parserMembers()>
>>
```

```

parserCtorBody() ::= <%
<super.parserCtorBody()><\n>
TreeAdaptor =
<if(actions.(actionScope).treeAdaptorInitializer)>
<actions.(actionScope).treeAdaptorInitializer>
<else>
new <actions.(actionScope).treeAdaptorType; null="CommonTreeAdaptor">()
<end>
;
%>

/** Add an adaptor property that knows how to build trees */
parserMembers() ::= <<
private <treeAdaptorType()> adaptor;

public <treeAdaptorType()> TreeAdaptor
{
get
{
return adaptor;
}

set
{
this.adaptor = value;
<grammar.directDelegates: {g|<g:delegateName()>.TreeAdaptor = this.adaptor;}>
}
}
>>

treeAdaptorType() ::= <<
<actions.(actionScope).treeAdaptorType; null="ITreeAdaptor">
>>

ruleReturnBaseType() ::= <%
Ast<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope\<<ASTLabelType>, <labelType>>
%>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
<ASTLabelType> root_0 = default(<ASTLabelType>);<\n>
>>

ruleLabelDefs() ::= <<
<super.ruleLabelDefs()>
<[ruleDescriptor.tokenLabels,ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]

```

```

: {it|<ASTLabelType> <it.label.text>_tree = default(<ASTLabelType>);}; separator="\n">
<ruleDescriptor.tokenListLabels: {it|<ASTLabelType> <it.label.text>_tree = default(<ASTLabelType>);};
separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites
: {it|RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>");}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
: {it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule <it>");};
separator="\n">
>>

```

```

/** When doing auto AST construction, we must define some variables;
* These should be turned off if doing rewrites. This must be a "mode"
* as a rule could have both rewrite and AST within the same alternative
* block.
*/

```

```

@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
root_0 = (<ASTLabelType>)adaptor.Nil();
<endif>
<endif>
<endif>
>>

```

// Tracking Rule Elements

```

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<token>.Add(<label>);<\n>
>>

```

```

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
* to the tracking list stream_ID for use in the rewrite.
*/
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefTrack(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

```

```

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<token>.Add(<label>);
>>

```

```

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<rule.name>.Add(<label>.Tree);
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<rule>.Add(<label>.Tree);
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

// R e w r i t e

rewriteCode(
alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
referencedWildcardLabels,
referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::= <<
<\n>{
// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">

```

```

// rule list labels: <referencedRuleListLabels; separator=", ">
// wildcard labels: <[referencedWildcardLabels,referencedWildcardListLabels]; separator=", ">
<if(backtracking)>
if (<actions.(actionScope).synpredgate>) {
<endif>
<prevRuleRootRef().Tree = root_0;
<rewriteCodeLabels()>
root_0 = (<ASTLabelType>)adaptor.Nil();
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER&&rewriteMode)>
<prevRuleRootRef().Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
if (<prevRuleRootRef().Tree != null)
input.ReplaceChildren(adaptor.GetParent(retval.Start), adaptor.GetChildIndex(retval.Start),
adaptor.GetChildIndex(_last), retval.Tree);
<endif>
<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER||!rewriteMode)>
<prevRuleRootRef().Tree = root_0;
<endif>
<if(backtracking)>
}
<endif>
}
>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
:{it|RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>",<it>)};
separator="\n"
>
<referencedTokenListLabels
:{it|RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>",<it>)};
separator="\n"
>
<referencedWildcardLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedWildcardListLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedRuleLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule

```

```

<it>",<it>!=null?<it>.Tree:null);};
    separator="\n"
>
<referencedRuleListLabels
  :{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"token <it>",<it>);};
  separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
// <fileName>:<description>
if (<referencedElementsDeep:{el | stream_<el>.HasNext}; separator="||">)
{
  <alt>
}
<referencedElementsDeep:{el | stream_<el>.Reset();<n>}>
>>

rewriteClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
// <fileName>:<description>
while ( <referencedElements:{el | stream_<el>.HasNext}; separator="||"> )
{
  <alt>
}
<referencedElements:{el | stream_<el>.Reset();<n>}>
>>

rewritePositiveClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
if (!(<referencedElements:{el | stream_<el>.HasNext}; separator="||">))
{

```

```

    throw new RewriteEarlyExitException();
}
while ( <referencedElements:{el | stream_<el>.HasNext}; separator="||"> )
{
    <alt>
}
<referencedElements:{el | stream_<el>.Reset();<\n>}>
>>

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>)
{
    <a.alt>
}
<else>
{
    <a.alt>
}
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
    <ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.Nil();
    <root:rewriteElement()>
    <children:rewriteElement()>
    adaptor.AddChild(root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <%
<@pregen()>
DebugLocation(<e.line>, <e.pos>);<\n>
<e.el>
%>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,args,terminalOptions={}) ::= <<
adaptor.AddChild(root_<treeLevel>, <createRewriteNodeFromElement(...)>);<\n>
>>

```



```

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextNode());<\n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextNode());<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,args,terminalOptions={ }) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<createRewriteNodeFromElement(...)>,
root_<treeLevel>);<\n>
>>

rewriteImaginaryTokenRef(args,token,elementIndex,terminalOptions={ }) ::= <<
adaptor.AddChild(root_<treeLevel>, <createImaginaryNode(tokenType=token, ...)>);<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,elementIndex,terminalOptions={ }) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<createImaginaryNode(tokenType=token, ...)>,
root_<treeLevel>);<\n>
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<rule>.NextTree());<\n>

```

```

>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<rule>.NextNode(), root_<treeLevel>);<\n>
>>

rewriteNodeAction(action) ::= <<
adaptor.AddChild(root_<treeLevel>, <action>);<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<action>, root_<treeLevel>);<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

rewriteWildcardLabelRef(label) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

createImaginaryNode(tokenType,args,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
<! new MethodNode(IDLabel, args) !>
new <terminalOptions.node><(tokenType><if(args)>, <args; separator=", "><endif>)
<else>
(<ASTLabelType>)adaptor.Create(<tokenType>, <args; separator=", "><if(!args)>"<tokenType>"<endif>)
<endif>
%>

createRewriteNodeFromElement(token,args,terminalOptions={ }) ::= <%

```

```

<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.NextToken())<if(args)>, <args; separator=", "><endif>
<else>
<if(args)> <! must create new node from old !>
adaptor.Create(<token>, <args; separator=", ">)
<else>
stream_<token>.NextNode()
<endif>
<endif>
%>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp2/AST.stg
```

No license file was found, but licenses were detected in source scan.

[The "BSD license"]

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python3/Dbg.stg
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python/AST.stg
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python/Dbg.stg
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python3/AST.stg
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Java/ASTParser.stg
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/serialize.g
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ActionScript/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

[The "BSD license"]

Copyright (c) 2010 Matthew Lloyd

<http://linkedin.com/in/matthewl>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
scalaTypeInitMap ::= [  
  "Int": "0",  
  "Long": "0",  
  "Float": "0.0f",  
  "Double": "0.0",  
  "Boolean": "false",  
  "Byte": "0",  
  "Short": "0",  
  "Char": "0",  
  default: "null" // anything other than an atomic type  
]
```

```
/** The overall file structure of a recognizer; stores methods for rules  
 * and cyclic DFAs plus support code.
```

```
*/
```

```
outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,  
  docComment, recognizer,  
  name, tokens, tokenNames, rules, cyclicDFAs,  
  bitsets, buildTemplate, buildAST, rewriteMode, profile,  
  backtracking, synpreds, memoize, numRules,  
  fileName, ANTLRVersion, generatedTimestamp, trace,  
  scopes, superClass, literals) ::=
```

```

<<
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
<actions.(actionScope).header>

<@imports>
import org.antlr.runtime._
<if(TREE_PARSER)>
import org.antlr.runtime.tree._
<endif>
<@end>

<docComment>
<recognizer>
>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode, labelType="CommonToken",
    superClass="Lexer") ::= <<
object <grammar.recognizerName> {
    <tokens:{it | val <it.name> = <it.type>}; separator="\n">

    <cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>
}

class <grammar.recognizerName>(input: CharStream, state<grammar.delegates:{g|, <g.recognizerName>
<g.delegateName()>}: RecognizerSharedState) extends <@superClassName><superClass><@end>(input,
state<grammar.delegates:{g|, <g.recognizerName> <g.delegateName()>}>) {
    import <grammar.recognizerName>._
    <actions.lexer.members>

    // delegates
    <grammar.delegates:
        {g|<g.recognizerName> <g.delegateName()>}; separator="\n">
    // delegators
    <grammar.delegates:
        {g|<g.recognizerName> <g.delegateName()>}; separator="\n">
    <last(grammar.delegates):{g|public <g.recognizerName> gParent;}>

    <scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScope()><endif>}>

    def this(input<grammar.delegates:{g|, <g.recognizerName> <g.delegateName()>}>: CharStream) =
        this(input, new RecognizerSharedState()<grammar.delegates:{g|, <g.delegateName()>}>)

    <if(memoize)>
    <if(grammar.grammarIsRoot)>
        state.ruleMemo = new Array[java.util.Map[_,_]](<numRules>+1)<\n> <! index from 1..n !>
    <endif>
    <endif>

    <grammar.directDelegates:

```

```

    {g|<g:delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|,
<p:delegateName()>>, this}); separator="\n">
    <grammar.delegators:
    {g|this.<g:delegateName()> = <g:delegateName()>}; separator="\n">
    <last(grammar.delegators):{g|gParent = <g:delegateName()>}>

override def getGrammarFileName = "<fileName>"

<if(filterMode)>
    <filteringNextToken()>
<endif>
    <rules; separator="\n\n">

    <synpreds:{p | <lexerSynpred(p)>}>
    <cyclicDFAs:{dfa | private val dfa<dfa.decisionNumber> = new
<grammar.recognizerName>.DFA<dfa.decisionNumber>(this)}; separator="\n">
}
>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */
filteringNextToken() ::= <<
override def nextToken(): Token = {
    while (true) {
        if ( input.LA(1)==CharStream.EOF ) {
            var eof: Token = new CommonToken((CharStream)input,Token.EOF,
                Token.DEFAULT_CHANNEL,
                input.index(),input.index())
            eof.setLine(getLine())
            eof.setCharPositionInLine(getCharPositionInLine())
            return eof
        }
        state.token = null
state.channel = Token.DEFAULT_CHANNEL
state.tokenStartCharIndex = input.index()
state.tokenStartCharPositionInLine = input.getCharPositionInLine()
state.tokenStartLine = input.getLine()
state.text = null
try {
    val m = input.mark()
    state.backtracking=1 <! means we won't throw slow exception !>
    state.failed=false
    mTokens()
    state.backtracking=0

```

```

    <! mTokens backtracks with synpred at backtracking==2
      and we set the synpredgate to allow actions at level 1. !>
    if ( state.failed ) {
      input.rewind(m)
      input.consume() <! advance one char and try again !>
    }
    else {
      emit()
      return state.token
    }
  }
}
catch {
  case re: RecognitionException =>
    // shouldn't happen in backtracking mode, but...
    reportError(re)
    recover(re)
  }
}
}

override def memoize(input: IntStream,
  ruleIndex: Int,
  ruleStartIndex: Int) = {
  if ( state.backtracking>1 ) super.memoize(input, ruleIndex, ruleStartIndex)
}

override def alreadyParsedRule(input: IntStream, ruleIndex: Int):Boolean {
  if ( state.backtracking>1 ) return super.alreadyParsedRule(input, ruleIndex)
  return false
}
}
>>

actionGate() ::= "state.backtracking==0"

filteringActionGate() ::= "state.backtracking==1"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
  bitsets, inputStreamType, superClass,
  labelType, members, rewriteElementType,
  filterMode, ASTLabelType="Object") ::= <<
object <grammar.recognizerName> {
<if(grammar.grammarIsRoot)>
  val tokenNames = Array(
    "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ">
  )<\n>
<endif>

```

```

<tokens:{it | val <it.name> = <it.type>}; separator="\n">

<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

<bitsets:{it | <bitset(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},
    words64=it.bits)>>
}

class <grammar.recognizerName>(input: <inputStreamType>, state<grammar.delegators:{g|, <g.recognizerName>
<g.delegateName()>>: RecognizerSharedState) extends <@superClassName><superClass><@end>(input, state) {
    import <grammar.recognizerName>._
    // delegates
    <grammar.delegators:
        {g|public <g.recognizerName> <g.delegateName()>}; separator="\n">
    // delegators
    <grammar.delegators:
        {g|public <g.recognizerName> <g.delegateName()>}; separator="\n">
    <last(grammar.delegators):{g|public <g.recognizerName> gParent;}>

    <scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScope()><endif>}>

    <@members>
    <! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
    def this(input<grammar.delegators:{g|, <g.recognizerName> <g.delegateName()>>: <inputStreamType>) =
        this(input, new RecognizerSharedState()<grammar.delegators:{g|, <g.delegateName()>>))

    <parserCtorBody()>
    <grammar.directDelegates:
        {g|<g.delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|,
<p:delegateName()>>, this)}; separator="\n">
        <grammar.indirectDelegates:{g | <g.delegateName()> = <g.delegator:delegateName()>.<g.delegateName()>};
separator="\n">
        <last(grammar.delegators):{g|gParent = <g.delegateName()>}>
    <@end>

    override def getTokenNames: Array[String] = tokenNames
    override def getGrammarFileName = "<fileName>"

    <members>

    <rules; separator="\n\n">

    <! generate rule/method definitions for imported rules so they
    appear to be defined in this recognizer. !>
    // Delegated rules
    <grammar.delegatedRules:{ruleDescriptor|
        @throws(classOf[RecognitionException])
        def <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope()>): <returnType()> = \{

```



```

<if(ruleDescriptor.hasReturnValue)>return
<endif><ruleDescriptor.grammar:delegateName(>>.<ruleDescriptor.name><ruleDescriptor.parameterScope.attributes: {a|<a.name>}; separator=", "> \} }>; separator="\n">

    <synpreds: {p | <synpred(p)>}>

    <cyclicDFAs: {dfa | private val dfa<dfa.decisionNumber> = new
<grammar.recognizerName>.DFA<dfa.decisionNumber>(this)}; separator="\n">
    }
>>

parserCtorBody() ::= <<
<if(memoize)>
<if(grammar.grammarIsRoot)>
this.state.ruleMemo = new Array[java.util.Map[_,_]](<length(grammar.allImportedRules)>+1)<\n> <! index from
1..n !>
<endif>
<endif>
<grammar.delegators:
{g|this.<g:delegateName(> = <g:delegateName(>)}; separator="\n">
>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,
    ASTLabelType="Object", superClass="Parser", labelType="Token",
    members={<actions.parser.members>}) ::= <<
<genericParser(inputStreamType="TokenStream", rewriteElementType="Token", ...)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
    numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="Object",
superClass={<if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif>},
members={<actions.treeparser.members>}
) ::= <<
<genericParser(inputStreamType="TreeNodeStream", rewriteElementType="Node", ...)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<

```

```

// $ANTLR start <ruleName>
@throws(classOf[RecognitionException])
def <ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope(>): Unit = {
    <ruleLabelDefs(>
<if(trace)>
    traceIn("<ruleName>_fragment", <ruleDescriptor.index>)
    try {
        <block>
    }
    finally {
        traceOut("<ruleName>_fragment", <ruleDescriptor.index>);
    }
<else>
    <block>
<endif>
}
// $ANTLR end <ruleName>
>>

```

```

synpred(name) ::= <<
final def <name>(): Boolean = {
    state.backtracking+=1
    <@start(>
    val start = input.mark()
    try {
        <name>_fragment() // can never throw exception
    } catch {
        case re: RecognitionException =>
            System.err.println("impossible: "+re)
    }
    val success = !state.failed
    input.rewind(start)
    <@stop(>
    state.backtracking-=1
    state.failed=false
    success
}<\n>
>>

```

```

lexerSynpred(name) ::= <<
<synpred(name)>
>>

```

```

ruleMemoization(name) ::= <<
<if(memoize)>
if ( state.backtracking>0 && alreadyParsedRule(input, <ruleDescriptor.index> ) ) { return <ruleReturnValue(> }
<endif>
>>

```

```

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>if (state.failed) return <ruleReturnValue()><endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (state.backtracking>0) { state.failed=true; return <ruleReturnValue()>}<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(scope=ruleDescriptor.returnScope)>

// $ANTLR start "<ruleName>"
// <fileName>:<description>
@throws(classOf[RecognitionException])
final def <ruleName>(<ruleDescriptor.parameterScope:parameterScope>): <returnType()> = {
  <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>)<endif>
  <ruleScopeSetUp()>
  <ruleDeclarations()>
  <ruleLabelDefs()>
  <ruleDescriptor.actions.init>
  <@preamble()>
  try {
    <ruleMemoization(name=ruleName)>
    <block>
    <ruleCleanUp()>
    <(ruleDescriptor.actions.after):execAction()>
  }
  <if(exceptions)>
    <exceptions:{e|<catch(decl=e.decl,action=e.action)><\n>}>
  <else>
  <if(!emptyRule)>
  <if(actions.(actionScope).rulecatch)>
    <actions.(actionScope).rulecatch>
  <else>
  catch {
    case re: RecognitionException =>
      reportError(re)
      recover(input,re)
  <@setErrorReturnValue()>
  }<\n>
  <endif>

```

```

<endif>
<endif>
  finally {
    <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
    <memoize()>
    <ruleScopeCleanUp()>
    <finally>
  }
  <@postamble()>
  return <ruleReturnValue()>
}
// $ANTLR end "<ruleName>"
>>

catch(decl,action) ::= <<
catch (<e.decl>) {
  <e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
val retval = new <returnType()>()
retval.start = input.LT(1)<\n>
<else>
<ruleDescriptor.returnScope.attributes: { a |
var <a.name>: <a.type> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>
}>
<endif>
<if(memoize)>
val <ruleDescriptor.name>_startIndex = input.index()
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: {it | <it>_stack.push(new <it>_scope()); separator="\n">
<ruleDescriptor.ruleScope: {it | <it.name>_stack.push(new <it.name>_scope()); separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: {it | <it>_stack.pop(); separator="\n">
<ruleDescriptor.ruleScope: {it | <it.name>_stack.pop(); separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]

```

```

: {it | var <it.label.text>: <labelType> = null}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: {it | var list_<it.label.text>: java.util.List=null}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels:{ll|var <ll.label.text>: RuleReturnScope = null}; separator="\n">
>>

```

```

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | var <it.label.text>: <labelType>=null}; separator="\n"
>
<ruleDescriptor.charLabels:{it | int <it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]
: {it | var list_<it.label.text>: java.util.List=null}; separator="\n"
>
>>

```

```

ruleReturnValue() ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
>>

```

```

ruleCleanup() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.stop = input.LT(-1)<\n>
<endif>
<endif>
>>

```

```

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( state.backtracking>0 ) { memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex) }
<endif>
<endif>

```

```
>>
```

```
/** How to generate a rule in the lexer; naked blocks are used for  
* fragment rules.  
*/
```

```
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<  
// $ANTLR start "<ruleName>"  
@throws(classOf[RecognitionException])  
final def m<ruleName>(<ruleDescriptor.parameterScope:parameterScope()>): Unit = {  
  <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>)<endif>  
  <ruleScopeSetUp()>  
  <ruleDeclarations()>  
  try {  
<if(nakedBlock)>  
    <ruleMemoization(name=ruleName)>  
    <lexerRuleLabelDefs()>  
    <ruleDescriptor.actions.init>  
    try <block><\n>  
<else>  
    var _type = <ruleName>  
    var _channel = BaseRecognizer.DEFAULT_TOKEN_CHANNEL  
    <ruleMemoization(name=ruleName)>  
    <lexerRuleLabelDefs()>  
    <ruleDescriptor.actions.init>  
    try <block>  
    <ruleCleanUp()>  
    state.`type` = _type  
    state.channel = _channel  
    <(ruleDescriptor.actions.after):execAction()>  
<endif>  
  }  
  finally {  
    <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>)<endif>  
    <ruleScopeCleanUp()>  
    <memoize()>  
  }  
}  
// $ANTLR end "<ruleName>"  
>>
```

```
/** How to generate code for the implicitly-defined lexer grammar rule  
* that chooses between lexer rules.  
*/
```

```
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<  
@throws(classOf[RecognitionException])  
def mTokens(): Unit = {  
  <block><\n>  
}
```

>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */

block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

var alt<decisionNumber> = <maxAlt>

<decls>

<@predecision()>

<decision>

<@postdecision()>

<@prebranch()>

alt<decisionNumber> match {

<alts:{a | <altSwitchCase(i,a)>}>

case _ =>

}

<@postbranch()>

>>

/** A rule block with multiple alternatives */

ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

var alt<decisionNumber> = <maxAlt>

<decls>

<@predecision()>

<decision>

<@postdecision()>

alt<decisionNumber> match {

<alts:{a | <altSwitchCase(i,a)>}>

case _ =>

}

>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

// <fileName>:<description>

<decls>

<@prealt()>

<alts>

<@postalt()>

>>

/** A special case of a (...) subrule with a single alternative */

blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

// <fileName>:<description>

<decls>

<@prealt()>

<alts>

```

<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
var cnt<decisionNumber>: Int = 0
<decls>
<@preloop()>
var loop<decisionNumber>_quitflag = false
while (!loop<decisionNumber>_quitflag) {
  var alt<decisionNumber>:Int = <maxAlt>
  <@predecision()>
  <decision>
  <@postdecision()>
  alt<decisionNumber> match {
    <alts:{a | <altSwitchCase(i,a)>}>
  }
  case _ =>
    if ( cnt<decisionNumber> >= 1 ) loop<decisionNumber>_quitflag = true
    else {
      <ruleBacktrackFailure()>
      val eee = new EarlyExitException(<decisionNumber>, input)
      <@earlyExitException()>
      throw eee
    }
  }
  cnt<decisionNumber>+=1
}
<@postloop()>
>>

```

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

```

```

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
<decls>
<@preloop()>
var loop<decisionNumber>_quitflag = false
while (!loop<decisionNumber>_quitflag) {
  var alt<decisionNumber>:Int = <maxAlt>
  <@predecision()>
  <decision>
  <@postdecision()>
  alt<decisionNumber> match {
    <alts:{a | <altSwitchCase(i,a)>}>
  }
}

```



```

    case _ => loop<decisionNumber>_quitflag = true
    }
}
<@postloop()>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation
 * so we can just use the normal block template
 */
optionalBlock ::= block

optionalBlockSingleAlt ::= block

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */
altSwitchCase(altNum, alt) ::= <<
case <altNum> =>
    <@prealt()>
    <alt>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
}
>>

/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(e) ::= <<
<@prematch()>
<e.el><\n>
>>

```

```

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(label)><label>=<endif>`match`(input,<token>,FOLLOW_<token>_in_<ruleName><elementIndex>)<if(label)>
.asInstanceOf[<labelType>]<endif>
<checkRuleBacktrackFailure()>
>>

```

```

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

```

```

listLabel(label,elem) ::= <<
if (list_<label>==null) list_<label>=new java.util.ArrayList()
list_<label>.add(<elem>)<\n>
>>

```

```

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = input.LA(1)<\n>
<endif>
`match`(<char>)
<checkRuleBacktrackFailure()>
>>

```

```

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = input.LA(1)<\n>
<endif>
matchRange(<a>,<b>); <checkRuleBacktrackFailure()>
>>

```

```

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,terminalOptions,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label>= input.LA(1)<\n>
<else>
<label>=input.LT(1).asInstanceOf[<labelType>]<\n>
<endif>
<endif>
<endif>
if ( <s> ) {
    input.consume()
    <postmatchCode>
}

```

```

<if(!LEXER)>
  state.errorRecovery=false<\n>
<endif>
  <if(backtracking)>state.failed=false<endif>
}
else {
  <ruleBacktrackFailure()>
  val mse = new MismatchedSetException(null,input)
  <@mismatchedSetException()>
<if(LEXER)>
  recover(mse)
  throw mse
<else>
  throw mse
  <! use following code to make it recover inline; remove throw mse;
  recoverFromMismatchedSet(input,mse,FOLLOW_set_in_<ruleName><elementIndex>)
  !>
<endif>
}<\n>
>>

```

```

matchRuleBlockSet ::= matchSet

```

```

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

```

```

/** Match a string literal */

```

```

lexerStringRef(string,label,elementIndex="0") ::= <<
<if(label)>
val <label>Start = getCharIndex()
`match`(<string>)
<checkRuleBacktrackFailure()>
val <label>StartLine<elementIndex> = getLine()
val <label>StartCharPos<elementIndex> = getCharPositionInLine()
<label> = new <labelType>(input, Token.INVALID_TOKEN_TYPE, Token.DEFAULT_CHANNEL, <label>Start,
getCharIndex()-1)
<label>.setLine(<label>StartLine<elementIndex>)
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
<else>
`match`(<string>)
<checkRuleBacktrackFailure()><\n>
<endif>
>>

```

```

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>

```

```

<label>=input.LT(1).asInstanceOf[<labelType>]<\n>
<endif>
matchAny(input)
<checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = input.LA(1)<\n>
<endif>
matchAny()
<checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabel(elem=label,...)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
pushFollow(FOLLOW_<rule.name>_in_<ruleName><elementIndex>)
<if(label)><label>=<endif><if(scope)><scope.delegateName()>.<endif><rule.name>(<args; separator=", ">)<\n>
state._fsp-=1
<checkRuleBacktrackFailure()>
>>

/** ids+=1 */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.

```

```

* Now though you can access full rule descriptor stuff.
*/
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
val <label>Start<elementIndex> = getCharIndex()
val <label>StartLine<elementIndex> = getLine()
val <label>StartCharPos<elementIndex> = getCharPositionInLine()
<if(scope)><scope.delegateName().<endif>m<rule.name>(<args; separator=", ">)
<checkRuleBacktrackFailure()>
<label> = new <labelType>(input, Token.INVALID_TOKEN_TYPE, Token.DEFAULT_CHANNEL,
<label>Start<elementIndex>, getCharIndex()-1)
<label>.setLine(<label>StartLine<elementIndex>)
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
<else>
<if(scope)><scope.delegateName().<endif>m<rule.name>(<args; separator=", ">)
<checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
val <label>Start<elementIndex> = getCharIndex()
val <label>StartLine<elementIndex> = getLine()
val <label>StartCharPos<elementIndex> = getCharPositionInLine()
`match`(EOF)
<checkRuleBacktrackFailure()>
val <label> = new <labelType>(input, EOF, Token.DEFAULT_CHANNEL, <label>Start<elementIndex>,
getCharIndex()-1)
<label>.setLine(<label>StartLine<elementIndex>)
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
<else>
`match`(EOF)
<checkRuleBacktrackFailure()>
<endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg()>"
recRuleArg()             ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0=${<ruleName>}_primary.tree;"

```

```
recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>";
```

```
/** match ^(root children) in tree parser */
```

```
tree(root, actionsAfterRoot, children, nullableChildList,  
    enclosingTreeLevel, treeLevel) ::= <<
```

```
<root:element()>
```

```
<actionsAfterRoot:element()>
```

```
<if(nullableChildList)>
```

```
if ( input.LA(1)==Token.DOWN ) {
```

```
    `match`(input, Token.DOWN, null)
```

```
    <checkRuleBacktrackFailure()>
```

```
    <children:element()>
```

```
    `match`(input, Token.UP, null)
```

```
    <checkRuleBacktrackFailure()>
```

```
}
```

```
<else>
```

```
`match`(input, Token.DOWN, null)
```

```
<checkRuleBacktrackFailure()>
```

```
<children:element()>
```

```
`match`(input, Token.UP, null)
```

```
<checkRuleBacktrackFailure()>
```

```
<endif>
```

```
>>
```

```
/** Every predicate is used as a validating predicate (even when it is
```

```
* also hoisted into a prediction expression).
```

```
*/
```

```
validateSemanticPredicate(pred,description) ::= <<
```

```
if ( !( <evalPredicate(...) > ) ) {
```

```
    <ruleBacktrackFailure()>
```

```
    throw new FailedPredicateException(input, "<ruleName>", "<description>")
```

```
}
```

```
>>
```

```
// F i x e d D F A (if-then-else)
```

```
dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
```

```
val LA<decisionNumber>_<stateNumber> = input.LA(<k>)<\n>
```

```
<edges; separator="\nelse ">
```

```
else {
```

```
<if(eotPredictsAlt)>
```

```
    alt<decisionNumber>=<eotPredictsAlt>
```

```
<else>
```

```
    <ruleBacktrackFailure()>
```

```
    val nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input)<\n>
```

```
    <@noViableAltException()>
```

```
    throw nvae<\n>
```

```
<endif>
```

```

}
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
val LA<decisionNumber>_<stateNumber> = input.LA(<k>)<\n>
<edges; separator="\nelse ">
>>

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
val LA<decisionNumber>_<stateNumber> = input.LA(<k>)<\n>
<edges; separator="\nelse "><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber>=<eotPredictsAlt> <! if no edges, don't gen ELSE !>
<else>
else {
  alt<decisionNumber>=<eotPredictsAlt>
}<\n>
<endif>
<endif>
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>"

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) {
  <targetState>
}
>>

// F i x e d D F A (switch case)

```

```

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
input.LA(<k>) match {
<edges; separator="\n">
case _ =>
<if(eotPredictsAlt)>
  alt<decisionNumber>=<eotPredictsAlt>
<else>
  <ruleBacktrackFailure()>
  val nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input)<\n>
  <@noViableAltException()>
  throw nvae<\n>
<endif>
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
input.LA(<k>) match {
  <edges; separator="\n">
  case _ =>
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
input.LA(<k>) match {
<edges; separator="\n"><\n>
case _ =>
<if(eotPredictsAlt)>
  alt<decisionNumber>=<eotPredictsAlt>;
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
case <labels:{it | <it>}>; separator=" | ">=>
{
  <targetState>
}
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */

```



```

dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = dfa<decisionNumber>.predict(input)
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
* Can't use hex as compiler translates them before compilation.
* These strings are split into multiple, concatenated strings.
* Java puts them back together at compile time thankfully.
* Java cannot handle large static arrays, so we're stuck with this
* encode/decode approach. See analysis and runtime DFA for
* the encoding methods.
*/
cyclicDFA(dfa) ::= <<
val DFA<dfa.decisionNumber>_eotS =
  "<dfa.javaCompressedEOT; wrap="+\n  \"">"
val DFA<dfa.decisionNumber>_eofS =
  "<dfa.javaCompressedEOF; wrap="+\n  \"">"
val DFA<dfa.decisionNumber>_minS =
  "<dfa.javaCompressedMin; wrap="+\n  \"">"
val DFA<dfa.decisionNumber>_maxS =
  "<dfa.javaCompressedMax; wrap="+\n  \"">"
val DFA<dfa.decisionNumber>_acceptS =
  "<dfa.javaCompressedAccept; wrap="+\n  \"">"
val DFA<dfa.decisionNumber>_specialS =
  "<dfa.javaCompressedSpecial; wrap="+\n  \"">}">"
val DFA<dfa.decisionNumber>_transitionS: Array[String] = Array(
  <dfa.javaCompressedTransition: {s|<s; wrap="+\n\"">" }; separator=",\n">
)

val DFA<dfa.decisionNumber>_eot: Array[Short] = DFA.unpackEncodedString(DFA<dfa.decisionNumber>_eotS)
val DFA<dfa.decisionNumber>_eof: Array[Short] = DFA.unpackEncodedString(DFA<dfa.decisionNumber>_eofS)
val DFA<dfa.decisionNumber>_min: Array[Char] =
DFA.unpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_minS)
val DFA<dfa.decisionNumber>_max: Array[Char] =
DFA.unpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_maxS)
val DFA<dfa.decisionNumber>_accept: Array[Short] =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_acceptS)
val DFA<dfa.decisionNumber>_special: Array[Short] =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_specialS)
val DFA<dfa.decisionNumber>_transition = new
Array[Array[Short]](DFA<dfa.decisionNumber>_transitionS.length)

for (i \<- DFA<dfa.decisionNumber>_transition.indices) {
  DFA<dfa.decisionNumber>_transition(i) =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_transitionS(i))
}

class DFA<dfa.decisionNumber> extends DFA {

```

```

def this(recognizer: BaseRecognizer) = {
  this()
  this.recognizer = recognizer
  this.decisionNumber = <dfa.decisionNumber>
  this.eot = DFA<dfa.decisionNumber>_eot
  this.eof = DFA<dfa.decisionNumber>_eof
  this.min = DFA<dfa.decisionNumber>_min
  this.max = DFA<dfa.decisionNumber>_max
  this.accept = DFA<dfa.decisionNumber>_accept
  this.special = DFA<dfa.decisionNumber>_special
  this.transition = DFA<dfa.decisionNumber>_transition
}
override def getDescription = "<dfa.description>"
<@errorMethod()>
<if(dfa.specialStateSTs)>
  @throws(classOf[NoViableAltException])
  override def specialStateTransition(s: Int, _input: IntStream):Int = {
    <if(LEXER)>
      val input = _input
    <endif>
    <if(PARSER)>
      val input = _input.asInstanceOf[TokenStream]
    <endif>
    <if(TREE_PARSER)>
      val input = _input.asInstanceOf[TreeNodeStream]
    <endif>
    val _s = s
    s match {
      <dfa.specialStateSTs:{state |
        case <i0> => <! compressed special state numbers 0..n-1 !>
          <state>}; separator="\n">
        case _ =>
          }
    }
  }
<if(backtracking)>
  if (state.backtracking>0) {state.failed=true; return -1}<\n>
<endif>
  val nvae = new NoViableAltException(getDescription(), <dfa.decisionNumber>, _s, input)
  error(nvae)
  throw nvae
}<\n>
<endif>
}<\n>
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */

```

```

cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
val LA<decisionNumber>_<stateNumber>: Int = input.LA(1)<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
val index<decisionNumber>_<stateNumber>: Int = input.index()
input.rewind()<\n>
<endif>
s = -1
<edges; separator="\nelse ">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>)<\n>
<endif>
if ( s>=0 ) return s
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) { s = <targetStateNumber>}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber><\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "<left>&&<right>"

orPredicates(operands) ::= "<operands; separator=\"||\">"

notPredicate(pred) ::= "!(<evalPredicate(pred,\""\")>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber>===<atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "input.LA(<k>)===<atom>"

```

```

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <<
(LA<decisionNumber>_<stateNumber> >= <lower> && LA<decisionNumber>_<stateNumber> \<= <upper>)
>>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(input.LA(<k>) >=<lower>
&& input.LA(<k>) \<= <upper>)"

setTest(ranges) ::= "<ranges; separator=\\\"\\\">"

// A T T R I B U T E S

globalAttributeScope(scope) ::= <<
<if(scope.attributes)>
class <scope.name>_scope {
  <scope.attributes:{it | var <it.name>: <it.type> = _}; separator="\n">
}
val <scope.name>_stack = new collection.mutable.Stack[<scope.name>_scope]<\n>
<endif>
>>

ruleAttributeScope(scope) ::= <<
<if(scope.attributes)>
class <scope.name>_scope {
  <scope.attributes:{it | var <it.name>: <it.type> = _}; separator="\n">
}
val <scope.name>_stack = new collection.mutable.Stack[<scope.name>_scope]<\n>
<endif>
>>

returnStructName(r) ::= "<r.name>_return"

returnType() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor:returnStructName()>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
Unit
<endif>
<endif>
>>

/** Generate the Java type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <<
<if(referencedRule.hasMultipleReturnValues)>

```

```

<referencedRule.name>_return
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
Unit
<endif>
<endif>
>>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<scalaTypeInitMap.(typeName)>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
var <label.label.text>: <ruleLabelType(referencedRule=label.referencedRule)> =
<initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))><\n>
>>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 * TODO(matthewlloyd): make this static
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
final class <ruleDescriptor:returnStructName()> extends
<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope {
  <scope.attributes:{it | var <it.name>: <it.type> = _}; separator="\n">
  <@ruleReturnMembers()>
}
<endif>
>>

parameterScope(scope) ::= <<
<scope.attributes:{it | <it.name>: <it.type>}; separator=", ">
>>

parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name> =<expr>"

```

```

scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack(<scope>_stack.size-<negIndex>-1).<attr.name>
<else>
<if(index)>
<scope>_stack(<index>).<attr.name>
<else>
<scope>_stack.top.<attr.name>
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack(<scope>_stack.size-<negIndex>-1).<attr.name> = <expr>
<else>
<if(index)>
<scope>_stack(<index>).<attr.name> = <expr>
<else>
<scope>_stack.top.<attr.name> = <expr>
<endif>
<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size(>)>0 && $function::name.equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
(if (<scope>!=null) <scope>.<attr.name> else <initValue(attr.type)>)
<else>
<scope>
<endif>
%>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
%>

```

```

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> =<expr>
<else>
<attr.name> =<expr>
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "(if (<scope>!=null) <scope>.getText() else null)"
tokenLabelPropertyRef_type(scope,attr) ::= "(if (<scope>!=null) <scope>.getType() else 0)"
tokenLabelPropertyRef_line(scope,attr) ::= "(if (<scope>!=null) <scope>.getLine() else 0)"
tokenLabelPropertyRef_pos(scope,attr) ::= "(if (<scope>!=null) <scope>.getCharPositionInLine() else 0)"
tokenLabelPropertyRef_channel(scope,attr) ::= "(if (<scope>!=null) <scope>.getChannel() else 0)"
tokenLabelPropertyRef_index(scope,attr) ::= "(if (<scope>!=null) <scope>.getTokenIndex() else 0)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "(if (<scope>!=null) Integer.valueOf(<scope>.getText()) else 0)"

ruleLabelPropertyRef_start(scope,attr) ::= "(if (<scope>!=null) <scope>.start.asInstanceOf[<labelType>] else null)"
ruleLabelPropertyRef_stop(scope,attr) ::= "(if (<scope>!=null) <scope>.stop.asInstanceOf[<labelType>] else null)"
ruleLabelPropertyRef_tree(scope,attr) ::= "(if (<scope>!=null) <scope>.tree.asInstanceOf[<ASTLabelType>] else
null)"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
(if (<scope>!=null) (input.getTokenStream().toString(
input.getTreeAdaptor().getTokenStartIndex(<scope>.start),
input.getTreeAdaptor().getTokenStopIndex(<scope>.start))) else null)
<else>
(if (<scope>!=null) input.toString(<scope>.start,<scope>.stop) else null)
<endif>
>>

ruleLabelPropertyRef_st(scope,attr) ::= "(if (<scope>!=null) <scope>.st else null)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::=
"(if (<scope>!=null) <scope>.getType() else 0)"
lexerRuleLabelPropertyRef_line(scope,attr) ::=

```

```

    "(if (<scope>!=null) <scope>.getLine() else 0)"
lexerRuleLabelPropertyRef_pos(scope,attr) ::=
    "(if (<scope>!=null) <scope>.getCharPositionInLine() else -1)"
lexerRuleLabelPropertyRef_channel(scope,attr) ::=
    "(if (<scope>!=null) <scope>.getChannel() else 0)"
lexerRuleLabelPropertyRef_index(scope,attr) ::=
    "(if (<scope>!=null) <scope>.getTokenIndex() else 0)"
lexerRuleLabelPropertyRef_text(scope,attr) ::=
    "(if (<scope>!=null) <scope>.getText() else null)"
lexerRuleLabelPropertyRef_int(scope,attr) ::=
    "(if (<scope>!=null) Integer.valueOf(<scope>.getText()) else 0)"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "(retval.start.asInstanceOf[<labelType>])"
rulePropertyRef_stop(scope,attr) ::= "(retval.stop.asInstanceOf[<labelType>])"
rulePropertyRef_tree(scope,attr) ::= "(retval.tree.asInstanceOf[<ASTLabelType>])"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
input.getTokenStream().toString(
    input.getTreeAdaptor().getTokenStartIndex(retval.start),
    input.getTreeAdaptor().getTokenStopIndex(retval.start))
<else>
input.toString(retval.start,input.LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "getText()"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(getCharIndex()-1)"
lexerRulePropertyRef_int(scope,attr) ::= "Integer.valueOf(<scope>.getText())"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>"

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
    <action>
}
}

```



```

<else>
<action>
<endif>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
val <name> = new BitSet(Array[Long](<words64:{it | <it>L};separator=","))<\n>
>>

codeFileExtension() ::= ".scala"

true_value() ::= "true"
false_value() ::= "false"

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Scala/Scala.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

[The "BSD license"]

Copyright (c) 2005-2006 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
@outputFile.imports() ::= <<
<@super.imports()>
<if(!TREE_PARSER)><! tree parser would already have imported !>
import org.antlr.runtime.tree.*;<\n>
<endif>
>>
```

```
@genericParser.members() ::= <<
<@super.members()>
<parserMembers()>
>>
```

```
/** Add an adaptor property that knows how to build trees */
parserMembers() ::= <<
protected var adaptor:TreeAdaptor = new CommonTreeAdaptor();<\n>
override public function set treeAdaptor(adaptor:TreeAdaptor):void {
    this.adaptor = adaptor;
    <grammar.directDelegates: {g|<g.delegateName().treeAdaptor = this.adaptor; }>
}
override public function get treeAdaptor():TreeAdaptor {
    return adaptor;
}
>>
```

```
@returnScope.ruleReturnMembers() ::= <<
<ASTLabelType> tree;
public function get tree():Object { return tree; }
>>
```

```
/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
var root_0:<ASTLabelType> = null;<\n>
>>
```

```
ruleLabelDefs() ::= <<
<super.ruleLabelDefs()>
<[ruleDescriptor.tokenLabels,ruleDescriptor.wildcardTreeLabels,
ruleDescriptor.wildcardTreeListLabels]:{it |var <it.label.text>_tree:<ASTLabelType>=null;}; separator="\n">
<ruleDescriptor.tokenListLabels: {it |var <it.label.text>_tree:<ASTLabelType>=null;}; separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites
: {it |var stream_<it>:RewriteRule<rewriteElementType>Stream=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>");}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
```

```

    :{it |var stream_<it>:RewriteRuleSubtreeStream=new RewriteRuleSubtreeStream(adaptor,"rule <it>");};
separator="\n">
>>

/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
root_0 = <ASTLabelType>(adaptor.nil());<\n>
<endif>
<endif>
<endif>
>>

// Tracking Rule Elements

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
 * to the tracking list stream_ID for use in the rewrite.
 */
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label,...)>
>>

/** rule when output=AST and tracking for rewrite */

```

```

ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule.name>.add(<label>.tree);
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabel(label, {<label>.tree})>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule>.add(<label>.tree);
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabel(label, {<label>.tree})>
>>

// R e w r i t e

rewriteCode(
alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
referencedWildcardLabels,
referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::=
<<

// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
// wildcard labels: <[referencedWildcardLabels,referencedWildcardListLabels]; separator=", ">
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {<\n>
<endif>
<prevRuleRootRef()>.tree = root_0;

```

```

<rewriteCodeLabels()>
root_0 = <ASTLabelType>(adaptor.nil());
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER)>
<if(rewriteMode)>
<prevRuleRootRef(>.tree = <ASTLabelType>(adaptor.rulePostProcessing(root_0));
input.replaceChildren(adaptor.getParent(retval.start),
    adaptor.getChildIndex(retval.start),
    adaptor.getChildIndex(_last),
    retval.tree);
<endif>
<endif>
<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER)>
<prevRuleRootRef(>.tree = root_0;
<else>
<if(!rewriteMode)>
<prevRuleRootRef(>.tree = root_0;
<endif>
<endif>
<if(backtracking)>
}
<endif>
>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
    :{it |var stream_<it>:RewriteRule<rewriteElementType>Stream=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>",<it>)};
    separator="\n"
>
<referencedTokenListLabels
    :{it |var stream_<it>:RewriteRule<rewriteElementType>Stream=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it> ", list_<it>)};
    separator="\n"
>
<referencedWildcardLabels
    :{it |var stream_<it>:RewriteRuleSubtreeStream=new RewriteRuleSubtreeStream(adaptor,"wildcard
<it>",<it>)};
    separator="\n"
>
<referencedWildcardListLabels
    :{it |var stream_<it>:RewriteRuleSubtreeStream=new RewriteRuleSubtreeStream(adaptor,"wildcard
<it> ",list_<it>)};
    separator="\n"
>
<referencedRuleLabels

```

```

    :{it |var stream_<it>:RewriteRuleSubtreeStream=new RewriteRuleSubtreeStream(adaptor,"rule
<it>",<it>!=null?<it>.tree:null);};
    separator="\n"
>
<referencedRuleListLabels
    :{it |var stream_<it>:RewriteRuleSubtreeStream=new RewriteRuleSubtreeStream(adaptor,"token
<it>",list_<it>);};
    separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
    alt,rewriteBlockLevel,
    referencedElementsDeep, // all nested refs
    referencedElements, // elements in immediately block; no nested blocks
    description) ::=
<<
// <fileName>:<description>
if ( <referencedElementsDeep:{el | stream_<el>.hasNext}; separator="||"> ) {
    <alt>
}
<referencedElementsDeep:{el | stream_<el>.reset();<n>}>
>>

rewriteClosureBlock(
    alt,rewriteBlockLevel,
    referencedElementsDeep, // all nested refs
    referencedElements, // elements in immediately block; no nested blocks
    description) ::=
<<
// <fileName>:<description>
while ( <referencedElements:{el | stream_<el>.hasNext}; separator="||"> ) {
    <alt>
}
<referencedElements:{el | stream_<el>.reset();<n>}>
>>

rewritePositiveClosureBlock(
    alt,rewriteBlockLevel,
    referencedElementsDeep, // all nested refs
    referencedElements, // elements in immediately block; no nested blocks
    description) ::=
<<
if ( !(<referencedElements:{el | stream_<el>.hasNext}; separator="||">) ) {
    throw new RewriteEarlyExitException();
}
}

```

```

}
while ( <referencedElements:{el | stream_<el>.hasNext}; separator="||"> ) {
  <alt>
}
<referencedElements:{el | stream_<el>.reset();<\n>}>
>>

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>) {
  <a.alt>
}<\n>
<else>
{
  <a.alt>
}<\n>
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
var root_<treeLevel>:<ASTLabelType> = <ASTLabelType>(adaptor.nil());
<root:rewriteElement()>
<children:rewriteElement()>
adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,terminalOptions,args) ::= <<
adaptor.addChild(root_<treeLevel>, <createRewriteNodeFromElement(...)>);<\n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());<\n>

```

```

>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>));<\n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,terminalOptions,args) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(<createRewriteNodeFromElement(...)>,
root_<treeLevel>));<\n>
>>

rewriteImaginaryTokenRef(args,token,terminalOptions,elementIndex) ::= <<
adaptor.addChild(root_<treeLevel>, <createImaginaryNode(token,terminalOptions,args)>);<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,terminalOptions,elementIndex) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(<createImaginaryNode(token,terminalOptions,args)>,
root_<treeLevel>));<\n>
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<rule>.nextTree());<\n>
>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(stream_<rule>.nextNode(), root_<treeLevel>));<\n>

```



```

>>

rewriteNodeAction(action) ::= <<
adaptor.addChild(root_<treeLevel>, <action>);<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(<action>, root_<treeLevel>));<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>));<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>));<\n>
>>

rewriteWildcardLabelRef(label) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

createImaginaryNode(tokenType,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
<! new MethodNode(IDLabel, args) !>
new <terminalOptions.node>(<tokenType><if(args)>, <args; separator=","><endif>)
<else>
<ASTLabelType>(adaptor.create(<tokenType>, <args; separator=","><if(!args)>"<tokenType>"<endif>))
<endif>
>>

createRewriteNodeFromElement(token,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.nextToken(<if(args)>, <args; separator=","><endif>))
<else>

```

```
<if(args)> <! must create new node from old !>
adaptor.create(<token>, <args; separator=", ">)
<else>
stream_<token>.nextNode()
<endif>
<endif>
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ActionScript/AST.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2010 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

```
THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

```
/** How to generate rules derived from left-recursive rules.
```

```
* These rely on recRuleDefArg(), recRuleAltPredicate(),
```

```
* recRuleArg(), recRuleSetResultAction(), recRuleSetReturnAction()
```

```
* templates in main language.stg
```

```
*/
```

```
group LeftRecursiveRules;
```

```

recRuleName(ruleName) ::= "<ruleName>_"
recPrimaryName(ruleName) ::= "<ruleName>_primary"

recRuleStart(ruleName, minPrec, userRetvals, userRetvalAssignments) ::= <<
<ruleName><if(userRetvals)> returns [<userRetvals>]<endif>
: <recRuleName(...)>[<minPrec>]
  <if(userRetvals)>
  {
    <userRetvalAssignments; separator="\n">
  }
  <endif>
;
>>

recRule(ruleName, precArgDef, argName, alts, setResultAction, buildAST,
  userRetvals, userRetvalAssignments) ::= <<
<recRuleName(...)>[<precArgDef>]<if(userRetvals)> returns [<userRetvals>]<endif>
: <recPrimaryName(...)>
  <if(buildAST)>
  {
    <setResultAction>
  }
  <endif>
  <if(userRetvals)>
  {
    <userRetvalAssignments; separator="\n">
  }
  <endif>
  ( options { backtrack=false;}
  : ( options { backtrack=false;}
    : <alts; separator="\n | ">
    )
  )*)
;
>>

recPrimaryRule(ruleName, alts, userRetvals) ::= <<
<recPrimaryName(...)><if(userRetvals)> returns [<userRetvals>]<endif>
options { backtrack=true;}
: <alts; separator="\n | ">
;
>>

recRuleAlt(alt, pred) ::= "{<pred>}?=> <alt>"

recRuleRef(ruleName, arg) ::= "<recRuleName(...)>[<arg>]"

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/LeftRecursiveRules.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2005-2008 Terence Parr
```

```
* All rights reserved.
```

```
*
```

```
* Conversion to C#:
```

```
* Copyright (c) 2008-2009 Sam Harwell, Pixel Mine, Inc.
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

```
* modification, are permitted provided that the following conditions
```

```
* are met:
```

```
* 1. Redistributions of source code must retain the above copyright
```

```
* notice, this list of conditions and the following disclaimer.
```

```
* 2. Redistributions in binary form must reproduce the above copyright
```

```
* notice, this list of conditions and the following disclaimer in the
```

```
* documentation and/or other materials provided with the distribution.
```

```
* 3. The name of the author may not be used to endorse or promote products
```

```
* derived from this software without specific prior written permission.
```

```
*
```

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```

```
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

```
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

```
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
```

```
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
```

```
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

```
/** Template overrides to add debugging to normal Java output;
```

```
* If ASTs are built, then you'll also get ASTDbg.stg loaded.
```

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
using Antlr.Runtime.Debug;
```

```
using IOException = System.IO.IOException;
```

```
>>
```

```
@genericParser.members() ::= <<
```

```
<if(grammar.grammarIsRoot)>
```

```
public static readonly string[] ruleNames =
```

```

new string[]
{
    "invalidRule", <grammar.allImportedRules:{rST | "<rST.name>" }; wrap="\n ", separator=", ">
};<\n>
<endif>
<if(grammar.grammarIsRoot)><! grammar imports other grammar(s) !>
int ruleLevel = 0;
public virtual int RuleLevel { get { return ruleLevel; } }
public virtual void IncRuleLevel() { ruleLevel++; }
public virtual void DecRuleLevel() { ruleLevel--; }
<if(profile)>
<ctorForProfilingRootGrammar()>
<else>
<ctorForRootGrammar()>
<endif>
<ctorForPredefinedListener()>
<else><! imported grammar !>
public int RuleLevel { get { return <grammar.delegators:{g| <g.delegateName()>>.RuleLevel; } } }
public void IncRuleLevel() { <grammar.delegators:{g| <g.delegateName()>>.IncRuleLevel(); } }
public void DecRuleLevel() { <grammar.delegators:{g| <g.delegateName()>>.DecRuleLevel(); } }
<ctorForDelegateGrammar()>
<endif>
<if(profile)>
public override bool AlreadyParsedRule( IIntStream input, int ruleIndex )
{
    int stopIndex = GetRuleMemoization(ruleIndex, input.Index);
    ((Profiler)dbg).ExamineRuleMemoization(input, ruleIndex, stopIndex,
    <grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
    return base.AlreadyParsedRule(input, ruleIndex);
}<\n>
public override void Memoize( IIntStream input, int ruleIndex, int ruleStartIndex )
{
    ((Profiler)dbg).Memoize(input, ruleIndex, ruleStartIndex,
    <grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
    base.Memoize(input, ruleIndex, ruleStartIndex);
}<\n>
<endif>
protected virtual bool EvalPredicate( bool result, string predicate )
{
    dbg.SemanticPredicate( result, predicate );
    return result;
}<\n>
>>

ctorForRootGrammar() ::= <<
<! bug: can't use <@super.members()> cut-n-paste instead !>
<! Same except we add port number and profile stuff if root grammar !>
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input )

```

```

: this( input, DebugEventSocketProxy.DefaultDebuggerPort, new RecognizerSharedState() )
{
}
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, int port,
RecognizerSharedState state )
: base( input, state )
{
<createListenerAndHandshake()>
<grammar.directDelegates:{g|<g:delegateName()> = new <g.recognizerName>( input, dbg, this.state,
this<grammar.delegates:{g|, <g:delegateName()>> }> );}; separator="\n">
<parserCtorBody()>
<@finally()>
}<\n>
>>

ctorForProfilingRootGrammar() ::= <<
<! bug: can't use <@super.members()> cut-n-paste instead !>
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input )
: this( input, new Profiler(null), new RecognizerSharedState() )
{
}
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg,
RecognizerSharedState state )
: base( input, dbg, state )
{
Profiler p = (Profiler)dbg;
p.setParser(this);
<grammar.directDelegates:
{g|<g:delegateName()> = new <g.recognizerName>( input, dbg, this.state, this<grammar.delegates:{g|,
<g:delegateName()>> }> );}; separator="\n">
<parserCtorBody()>
<@finally()>
}
}<\n>
>>

/** Basically we don't want to set any dbg listeners are root will have it. */
ctorForDelegateGrammar() ::= <<
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg,
RecognizerSharedState state<grammar.delegates:{g|, <g.recognizerName> <g:delegateName()>> }> )
: base( input, dbg, state )
{
<grammar.directDelegates:
{g|<g:delegateName()> = new <g.recognizerName>( input, this, this.state<grammar.delegates:{g|,
<g:delegateName()>> }> );}; separator="\n">
<parserCtorBody()>
}<\n>
>>

```

```

ctorForPredefinedListener() ::= <<
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg
)
<@superClassRef>: base( input, dbg, new RecognizerSharedState() )<@end>
{
<if(profile)>
Profiler p = (Profiler)dbg;
p.setParser(this);
<endif>
<grammar.directDelegates:{g|<g:delegateName()> = new <g.recognizerName>(input, dbg, this.state,
this<grammar.delegators:{g|, <g:delegateName()>}>)}; separator="\n">
<parserCtorBody()>
<@finally()>
}<\n>
>>

```

```

createListenerAndHandshake() ::= <<
<if(TREE_PARSER)>
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port, input.TreeAdaptor );<\n>
<else>
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port, null );<\n>
<endif>
DebugListener = proxy;
try
{
proxy.Handshake();
}
catch ( IOException ioe )
{
ReportError( ioe );
}
>>

```

```
@genericParser.superClassName() ::= "Debug<@super.superClassName()>"
```

```

/*
* Much of the following rules were merged into CSharp3.stg.
*/

```

```

@rule.preamble() ::= <<
if (RuleLevel == 0)
DebugListener.Commence();
IncRuleLevel();
>>

```

```

//@rule.preamble() ::= <<
//try
//{

```

```

// dbg.EnterRule( GrammarFileName, "<ruleName>" );
// if ( RuleLevel == 0 )
// {
//   dbg.Commence();
// }
// IncRuleLevel();
// dbg.Location( <ruleDescriptor.tree.line>, <ruleDescriptor.tree.charPositionInLine> );<\n>
//>>

@rule.postamble() ::= <<
DecRuleLevel();
if (RuleLevel == 0)
  DebugListener.Terminate();
>>

//@rule.postamble() ::= <<
//dbg.Location(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.charPositionInLine>);<\n>
//}
//finally
//{
//   dbg.ExitRule( GrammarFileName, "<ruleName>" );
//   DecRuleLevel();
//   if ( RuleLevel == 0 )
//   {
//     dbg.Terminate();
//   }
//}<\n>
//>>

//@insertSynpreds.start() ::= "dbg.BeginBacktrack( state.backtracking );"
//@insertSynpreds.stop() ::= "dbg.EndBacktrack( state.backtracking, success );"

// Common debug event triggers used by region overrides below

//enterSubRule() ::= <<
//try
//{
//   dbg.EnterSubRule( <decisionNumber> );<\n>
//>>

//exitSubRule() ::= <<
//}
//finally
//{
//   dbg.ExitSubRule( <decisionNumber> );
//}<\n>
//>>

//enterDecision() ::= <<

```



```

//try
//{
// dbg.EnterDecision( <decisionNumber> );<\n>
//>>

//exitDecision() ::= <<
//}
//finally
//{
// dbg.ExitDecision( <decisionNumber> );
//}<\n>
//>>

//enterAlt(n) ::= "dbg.EnterAlt( <n> );<\n>"

// Region overrides that tell various constructs to add debugging triggers

//@block.predecision() ::= "<enterSubRule()><enterDecision()>"

//@block.postdecision() ::= "<exitDecision()>"

//@block.postbranch() ::= "<exitSubRule()>"

//@ruleBlock.predecision() ::= "<enterDecision()>"

//@ruleBlock.postdecision() ::= "<exitDecision()>"

//@ruleBlockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

//@blockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

//@positiveClosureBlock.preloop() ::= "<enterSubRule()>"

//@positiveClosureBlock.postloop() ::= "<exitSubRule()>"

//@positiveClosureBlock.predecision() ::= "<enterDecision()>"

//@positiveClosureBlock.postdecision() ::= "<exitDecision()>"

//@positiveClosureBlock.earlyExitException() ::=
// "dbg.RecognitionException( eee<decisionNumber> );<\n>"

//@closureBlock.preloop() ::= "<enterSubRule()>"

//@closureBlock.postloop() ::= "<exitSubRule()>"

//@closureBlock.predecision() ::= "<enterDecision()>"

```

```

//@closureBlock.postdecision() ::= "<exitDecision>"

//@altSwitchCase.preatl() ::= "<enterAlt(n=i)>"

//@element.prematch() ::=
// "dbg.Location( <it.line>, <it.pos> );"

//@matchSet.mismatchedSetException() ::=
// "dbg.RecognitionException( mse );"

//@dfaState.noViableAltException() ::= "dbg.RecognitionException( nvae );"

//@dfaStateSwitch.noViableAltException() ::= "dbg.RecognitionException( nvae );"

dfaDecision(decisionNumber,description) ::= <<
//try
//{
// isCyclicDecision = true;
// <super.dfaDecision(...)>
//}
//catch ( NoViableAltException nvae )
//{
// dbg.RecognitionException( nvae );
// throw nvae;
//}
//>>

//@cyclicDFA.errorMethod() ::= <<
//public override void Error( NoViableAltException nvae )
//{
// ((DebugParser)recognizer).dbg.RecognitionException( nvae );
//}
//>>

/** Force predicate validation to trigger an event */
evalPredicate(pred,description) ::= <<
EvalPredicate(<pred>, "<description>")
>>

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/Dbg.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD license"]
Copyright (c) 2005-2006 Terence Parr
All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Template subgroup to add template rewrite output
```

```
* If debugging, then you'll also get STDbg.stg loaded.
```

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
import org.stringtemplate.v4.*;
```

```
import java.util.HashMap;
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturnMembers() ::= <<
```

```
/* ST returnScope.ruleReturnMembers -- empty */
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturn.memvars() ::= <<
```

```
ST *st;
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturn.properties() ::= <<
```

```
@property (retain) ST *st;
```

```
>>
```

```

/** Add this to each rule's return value struct */
@returnScope.ruleReturn.methodsDecl() ::= <<
- (ST *)getTemplate;
- (NSString *)toString;
>>

/** Add this to each rule's return value struct */
@returnScope.ruleReturn.synthesize() ::= <<
@synthesize st;
>>

/** Add this to each rule's return value struct */
@returnScope.ruleReturn.methods() ::= <<
- (ST *)getTemplate { return st; }
- (NSString *)toString { return st==nil?nil:[st render]; }
>>

@genericParser.members() ::= <<
<@super.members()>
STGroup *templateLib = [STGroup newSTGroup];

- (void)setTemplateLib:(STGroup *)aTemplateLib
{
    self.templateLib = aTemplateLib;
}

- (STGroup *)getTemplateLib
{
    return templateLib;
}
>>

@genericParserHeaderFile.memVars() ::= <<
<@super.memVars()>
/* ST genericParserHeaderFile.memVars -- empty now */
STGroup *templateLib; /* ST -- really a part of STAttrMap */
>>

@genericParserHeaderFile.properties() ::= <<
<@super.properties()>
/* ST genericParser.properties */
@property (retain, getter=getTemplateLib, setter=setTemplateLib:) STGroup *templateLib;
>>

@genericParserHeaderFile.methodsDecl() ::= <<
<@super.methodsDecl()>
/* ST genericParser.methodsDecl */
- init;

```

```

- (STGroup *) getTemplateLib;
- (void) setTemplateLib:(STGroup *)aTemplateLib;
@end
>>

```

```

@genericParser.synthesize() ::= <<
<@super.synthesize()>
/* ST genericParserImplementation.synthesize */
@synthesize templateLib;
>>

```

```

@genericParser.methods() ::= <<
<@super.methods()>
/* ST genericParser.methods */

```

```

- (STGroup *)getTemplateLib
{
    return templateLib;
}

```

```

- (void) setTemplateLib:(STGroup *)aTemplateLib
{
    templateLib = aTemplateLib;
}

```

```

>>

```

```

@genericParser.members() ::= <<
<@super.members()>
STGroup *templateLib = [STGroup newSTGroup];

```

```

- (STGroup *) getTemplateLib
{
    return templateLib;
}

```

```

- (void) setTemplateLib:(STGroup *) templateLib
{
    this.templateLib = templateLib;
}

```

```

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, { [<label> getTemplate]; })>
>>

```

```

rewriteTemplate(alts) ::= <<

```

```

// TEMPLATE REWRITE
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
  <alts:rewriteTemplateAlt(); separator="else ">
  <if(rewriteMode)><replaceTextInLine()><endif>
}
<else>
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>

replaceTextInLine() ::= <<
<if(TREE_PARSER)>
  [((TokenRewriteStream)input getTokenStream)]
  replaceFromIndex:[input getTreeAdaptor] getTokenStartIndex:retval.start]
  ToIndex:[input getTreeAdaptor] getTokenStopIndex:retval.start]
  Text:[retval.st render]];
<else>
  [((TokenRewriteStream)input)
  replaceFromIndex:[((Token)retval.start) getTokenIndex]
  ToIndex:[input LT:-1] getTokenIndex]
  Text:[retval.st render]];
<endif>
>>

rewriteTemplateAlt(alt) ::= <<
// <alt.description>
<if(alt.pred)>
if (<alt.pred>) {
  retval.st = <alt.alt>;
}<\n>
<else>
{
  retval.st = <alt.alt>;
}<\n>
<endif>
>>

rewriteEmptyTemplate(alts) ::= <<
nil;
>>

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template *after* having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded

```

```

* template.
*/
rewriteExternalTemplate(name,args) ::= <%
<if(args)><args:{ a | []}><endif>
[templateLib getInstanceOf:@"<name>"]
<if(args)><args:{ a | add:@"<a.name>" value:<a.value>}]><endif>;
%>

/** expr is a string expression that says what template to load */
rewriteIndirectTemplate(expr,args) ::= <%
<if(args)><args:{ a | []}><endif>
[templateLib getInstanceOf:<expr>];
<if(args)><args:{ a | add:@"<a.name>" value:<a.value>}]><endif>;
%>

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <%
<if(args)><args:{ a | []}><endif>
[ST newST:templateLib template:"<template>"]
<if(args)><args:{ a | add:@"<a.name>" <a.value>}]><endif>;
%>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
<action>
>>

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
[(<st>) setAttribute:@"<attrName>" value:<expr>];
>>

/** Translate %{stringExpr} */
actionStringConstructor(stringExpr) ::= <<
[ST newST:templateLib template:<stringExpr>];
>>

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ObjC/ST4ObjC.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

```

* [The "BSD license"]

```

```

* Copyright (c) 2011 Terence Parr

```

```

* All rights reserved.

```

```

*

```

```

* Conversion to C#:

```

* Copyright (c) 2011 Sam Harwell, Tunnel Vision Laboratories, LLC.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

```
dfa(decisionRanks,states,edges,rankdir,startState,useBox) ::= <<
digraph NFA {
<if(rankdir)>rankdir=<rankdir>;<endif>
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
nfa(decisionRanks,states,edges,rankdir,startState) ::= <<
digraph NFA {
rankdir=LR;
<decisionRanks; separator="\n">
<states; separator="\n">
<edges; separator="\n">
}
>>
```

```
decision-rank(states) ::= <<
{rank=same; rankdir=TB; <states; separator=" "; ">}
>>
```



```
edge(src,target,label,arrowhead) ::= <<
<src> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7, label = "<label>"<if(arrowhead)>, arrowhead =
<arrowhead><endif>];
>>
```

```
action-edge(src,target,label,arrowhead) ::= <<
<src> -> <target> [fontsize=11, fontname="Courier", arrowsize=.7, label = "<label>"<if(arrowhead)>, arrowhead =
<arrowhead><endif>];
>>
```

```
epsilon-edge(src,target,label,arrowhead) ::= <<
<src> -> <target> [fontname="Times-Italic", label = "e"];
>>
```

```
state(name,useBox) ::= <<
node [fontsize=11, shape = <if(useBox)>box<else>circle, fixedsize=true, width=.4<endif>]; <name>
>>
```

```
stopstate(name,useBox) ::= <<
node [fontsize=11, shape = <if(useBox)>polygon,sides=4,peripheries=2<else>doublecircle, fixedsize=true,
width=.6<endif>]; <name>
>>
```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/templates/dot/dot.stg
No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2012 Terence Parr

* Copyright (c) 2012 Sam Harwell

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/ToolSTGroupFile.java

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC

<http://www.temporal-wave.com>

<http://www.linkedin.com/in/jimidle>

All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
notice, this list of conditions and the following disclaimer in the
documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

```

* child, root
* token, set, rule, wildcard
*
* Each combination has its own template except that label/no label
* is combined into tokenRef, ruleRef, ...
*/
/** Add a variable to track last element matched */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
<ASTLabelType> _last;<\n>
<ASTLabelType> _first_0;<\n>
>>

/** Add a variable to track last element matched */
ruleInitializations() ::= <<
<super.ruleInitializations()>
_last = NULL;<\n>
_first_0 = NULL;<\n>
>>

/** What to emit when there is no rewrite rule. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= <<
<if(backtracking)>if ( BACKTRACKING ==0 ) {<endif>
<if(rewriteMode)>
retval.tree = (<ASTLabelType>)_first_0;
if ( ADAPTOR->getParent(ADAPTOR, retval.tree) != NULL && ADAPTOR->isNilNode(ADAPTOR,
ADAPTOR->getParent(ADAPTOR, retval.tree) ) )
{
    retval.tree = (<ASTLabelType>)ADAPTOR->getParent(ADAPTOR, retval.tree);
}
<endif>
<if(backtracking)>}<endif>
>>

/** match ^(root children) in tree parser; override here to
* add tree construction actions.
*/
tree(root, actionsAfterRoot, children, nullableChildList,
    enclosingTreeLevel, treeLevel) ::= <<
_last = (<ASTLabelType>)LT(1);
{
<ASTLabelType> _save_last_<treeLevel>;
<ASTLabelType> _first_<treeLevel>;
<if(!rewriteMode)>
<ASTLabelType> root_<treeLevel>;
<endif>

```

```

_save_last_<treeLevel> = _last;
_first_<treeLevel> = NULL;
<if(!rewriteMode)>
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->nilNode(ADAPTOR));
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if ( BACKTRACKING ==0 ) {<endif>
<if(root.el.rule)>
if ( _first_<enclosingTreeLevel> == NULL ) _first_<enclosingTreeLevel> = <root.el.label>.tree;
<else>
if ( _first_<enclosingTreeLevel> == NULL ) _first_<enclosingTreeLevel> = <root.el.label>;
<endif>
<if(backtracking)>}<endif>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( LA(1)==ANTLR3_TOKEN_DOWN ) {
    MATCHT(ANTLR3_TOKEN_DOWN, NULL);
    <children:element()>
    MATCHT(ANTLR3_TOKEN_UP, NULL);
}
<else>
MATCHT(ANTLR3_TOKEN_DOWN, NULL);
<children:element()>
MATCHT(ANTLR3_TOKEN_UP, NULL);
<endif>
<if(!rewriteMode)>
ADAPTOR->addChild(ADAPTOR, root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}<\n>
>>

```

```
// TOKEN AST STUFF
```

```

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.tokenRef(...)>
>>

```

```

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.tokenRef(...)>

```

```

<if(!rewriteMode)>
<if(backtracking)>if ( BACKTRACKING ==0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>New(<label>);
<else>
<label>_tree = (<ASTLabelType>)ADAPTOR->dupNode(ADAPTOR, <label>);
<endif>
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else>
<if(backtracking)>if ( BACKTRACKING ==0 ) {<endif>
if ( _first_<treeLevel> == NULL ) _first_<treeLevel> = <label>;
<if(backtracking)>}<endif>
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( BACKTRACKING == 0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>New(<label>);
<else>
<label>_tree = (<ASTLabelType>)ADAPTOR->dupNode(ADAPTOR, <label>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)ADAPTOR->becomeRoot(ADAPTOR, <label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.wildcard(...)>

```

```

<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)ADAPTOR->dupTree(ADAPTOR, <label>);
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
>>

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( BACKTRACKING == 0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>New(<label>);
<else>
<label>_tree = (<ASTLabelType>)ADAPTOR->dupNode(ADAPTOR, <label>);
<endif><\n>
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <label>_tree);
<if(backtracking)>}\}<endif>
<endif>
}
,...)>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
<noRewrite()> <! set return tree !>
>>

matchSetBang(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.matchSet(...)>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>New(<label>);
<else>
<label>_tree = (<ASTLabelType>)ADAPTOR->dupNode(ADAPTOR, <label>);

```

```

<endif>
root_<treeLevel> = (<ASTLabelType>)ADAPTOR->becomeRoot(ADAPTOR, <label>_tree, root_<treeLevel>);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
>>

// RULE REF AST

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )
{
<endif>
<if(!rewriteMode)>
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <label>.tree);
<else>
if ( _first_<treeLevel> == NULL ) _first_<treeLevel> = <label>.tree;
<endif>
<if(backtracking)>\}<endif>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<super.listLabelAST(elem=label,...)>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)LT(1);
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( ( <actions.(actionScope).synpredgate> ) ) <endif>root_<treeLevel> =
(<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, <label>.tree, root_<treeLevel>));
<endif>
>>

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<super.listLabelAST(elem=label,...)>
>>

/** rule when output=AST and tracking for rewrite */

```

```

ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
  _last = (<ASTLabelType>)LT(1);
  <super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  _last = (<ASTLabelType>)LT(1);
  <super.ruleRefTrackAndListLabel(...)>
>>
/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
  _last = (<ASTLabelType>)LT(1);
  <super.ruleRefRuleRootTrack(...)>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  _last = (<ASTLabelType>)LT(1);
  <super.ruleRefRuleRootTrackAndListLabel(...)>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,terminalOptions,scope) ::= <<
  <if(terminalOptions.node)>
  <terminalOptions.node>New(stream_<token>->nextNode(stream_<token>))
  <else>
  stream_<token>->nextNode(stream_<token>)
  <endif>
>>

ruleCleanUp() ::= <<
  <super.ruleCleanUp(...)>
  <if(backtracking)>
  if ( <actions.(actionScope).synpredgate> ) {<\n>
  <endif>
  <if(!ruleDescriptor.isSynPred)>
  retval.stop = LT(-1);<\n>
  <endif>
  retval.tree = (<ASTLabelType>)ADAPTOR->rulePostProcessing(ADAPTOR, root_0);
  <if(backtracking)>
  }
  <endif>
  <ruleDescriptor.allTokenRefsInAltsWithRewrites
  :{if (stream_<it> != NULL) stream_<it>->free(stream_<it>);}; separator="\n">

```



```
<ruleDescriptor.allRuleRefsInAltsWithRewrites
  :{if (stream_<it> != NULL) stream_<it>->free(stream_<it>);}; separator="\n">
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/C/ASTTreeParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
```

```
http://www.temporal-wave.com
```

```
http://www.linkedin.com/in/jimidle
```

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Templates for building ASTs during normal parsing.
```

```
*
```

```
* Deal with many combinations. Dimensions are:
```

```
* Auto build or rewrite
```

```
* no label, label, list label (label/no-label handled together)
```

```
* child, root
```

```
* token, set, rule, wildcard
```

```
*
```

* The situation is not too bad as rewrite (->) usage makes ^ and !
 * invalid. There is no huge explosion of combinations.
 */

```
@rule.setErrorReturnValue() ::= <<
retval.tree = (<ASTLabelType>)(ADAPTOR->errorNode(ADAPTOR, INPUT, retval.start, LT(-1), EXCEPTION));
>>
```

```
// TOKEN AST STUFF
```

```
/** ID and output=AST */
```

```
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<super.tokenRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)(ADAPTOR->create(ADAPTOR, <label>));
ADAPTOR->addChild(ADAPTOR, root_0, <label>_tree);
<if(backtracking)>}<endif>
>>
```

```
/** ID! and output=AST (same as plain tokenRef) */
```

```
tokenRefBang(token,label,elementIndex,terminalOptions) ::= "<super.tokenRef(...)>"
```

```
/** ID^ and output=AST */
```

```
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
<super.tokenRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = <createNodeFromToken(...)>;
root_0 = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, <label>_tree, root_0));
<if(backtracking)>}<endif>
>>
```

```
/** ids+=ID! and output=AST */
```

```
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<listLabel(elem=label,...)>
>>
```

```
/** label+=TOKEN when output=AST but not rewrite alt */
```

```
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>
```

```
/** Match label+=TOKEN^ when output=AST but not rewrite alt */
```

```
tokenRefRuleRootAndListLabel(token,label,terminalOptions,elementIndex) ::= <<
<tokenRefRuleRoot(...)>
```

```

<listLabel(elem=label,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<super.matchSet(postmatchCode={<if(backtracking)>if ( <actions.(actionScope).synpredgate> )
<endif>ADAPTOR->addChild(ADAPTOR, root_0, <createNodeFromToken(...)>};,...)>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,terminalOptions,postmatchCode) ::= "<super.matchSet(...)>"

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<if(label)>
<label>=(<labelType>)LT(1);<\n>
<endif>
<super.matchSet(postmatchCode={<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>root_0 =
(<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, <createNodeFromToken(...)>, root_0));},,...)>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>ADAPTOR->addChild(ADAPTOR, root_0,
<label>.tree);
>>

/** rule! is same as normal rule ref */

```

```

ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>root_0 = (<ASTLabelType>)(ADAPTOR-
>becomeRoot(ADAPTOR, <label>.tree, root_0));
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabelAST(...)>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabelAST(...)>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabelAST(...)>
>>

// WILDCARD AST

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<super.wildcard(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)(ADAPTOR->create(ADAPTOR, <label>));
ADAPTOR->addChild(ADAPTOR, root_0, <label>_tree);
<if(backtracking)>}<endif>
>>

wildcardBang(token,label,elementIndex,terminalOptions) ::= "<super.wildcard(...)>"

wildcardRuleRoot(label,elementIndex) ::= <<
<super.wildcard(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)(ADAPTOR->create(ADAPTOR, <label>));
root_0 = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, <label>_tree, root_0));
<if(backtracking)>}<endif>
>>

createNodeFromToken(label,terminalOptions) ::= <<

```

```

<if(terminalOptions.node)>
<terminalOptions.node>New(<label>) <! new MethodNode(IDLabel) !>
<else>
(<ASTLabelType>)(ADAPTOR->create(ADAPTOR, <label>))
<endif>
>>

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> )
{<\n>
<endif>
<if(!ruleDescriptor.isSynPred)>
retval.stop = LT(-1);<\n>
<endif>
retval.tree = (<ASTLabelType>)(ADAPTOR->rulePostProcessing(ADAPTOR, root_0));
ADAPTOR->setTokenBoundaries(ADAPTOR, retval.tree, retval.start, retval.stop);
<ruleDescriptor.allTokenRefsInAltsWithRewrites
: {it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>);}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
: {it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>);}; separator="\n">
<if(backtracking)>
}<\n>
<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/C/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2006 Kay Roepke
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

New style messages. This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the default format ANTLR uses.

*/

group antlr;

location(file, line, column) ::= "<file>:<line>:<column>:"

message(id, text) ::= "<id> <text>"

report(location, message, type) ::= "<type><message.id>: <location> <message.text>"

wantsSingleLineMessage() ::= "false"

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/templates/messages/formats/antlr.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2006 Kay Roepke 2010 Alan Condit

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to AST stuff. Dynamic inheritance

* hierarchy is set up as ASTDbg : AST : Dbg : Java by code generator.

*/

parserMembers() ::= <<

```
DebugTreeAdaptor *adaptor = [DebugTreeAdaptor newTreeAdaptor:(id)dbg Adaptor:[CommonTreeAdaptor
newTreeAdaptor]];
```

// fix this

- (void) setTreeAdaptor:(id<TreeAdaptor>)anAdaptor

{

```
    adaptor = [DebugTreeAdaptor newTreeAdaptor:dbg Adaptor:anAdaptor];
```

```
<if(grammar.grammarIsRoot)>
```

```
    adaptor = [DebugTreeAdaptor newTreeAdaptor:adaptor withDBG:dbg];
```

```
<else>
```

```
    adaptor = (DebugTreeAdaptor *)adaptor; // delegator sends dbg adaptor
```

```
<endif><\n>
```

```
    <grammar.directDelegates:{g[<g.delegateName()> setTreeAdaptor:adaptor];}>
```

}

- (id<TreeAdaptor>)getTreeAdaptor

{

```
    return adaptor;
```

```
}<\n>
```

>>

parserCtorBody() ::= <<

```
<super.parserCtorBody()>
```

>>

createListenerAndHandshake() ::= <<

```
DebugEventSocketProxy proxy =
```

```
    [DebugEventSocketProxy newDebugEventSocketProxy:self, port, <if(TREE_PARSER)>[input
getTreeAdaptor]<else>adaptor<endif>];
```

```
[self setDebugListener:proxy];
```

```
[self set<inputStreamType>:[Debug<inputStreamType> newDebug<inputStreamType>:input with:proxy]];
```

```
try {
```

```

    [proxy handshake];
  }
  @catch (IOException *ioe) {
    [self reportError:ioe];
  }
}
>>

```

```

@ctorForRootGrammar.finally() ::= <<
CommonTreeAdaptor *adap = [CommonTreeAdaptor newTreeAdaptor];
[self setTreeAdaptor:adap];
[proxy setTreeAdaptor:adap];
>>

```

```

@ctorForProfilingRootGrammar.finally() ::= <<
CommonTreeAdaptor *adap = [CommonTreeAdaptor newTreeAdaptor];
[self setTreeAdaptor:adap];
[proxy setTreeAdaptor:adap];
>>

```

```

@ctorForPredefinedListener.superClassRef() ::= @"super(input, dbg);"

```

```

@ctorForPredefinedListener.finally() ::= <<
<if(grammar.grammarIsRoot)> <! don't create new adaptor for delegates !>
CommonTreeAdaptor *adap = [CommonTreeAdaptor newTreeAdaptor];
[self setTreeAdaptor:adap];<\n>
<endif>
>>

```

```

@treeParserHeaderFile.superClassName ::= "DebugTreeParser"

```

```

@rewriteElement.pregen() ::= "[debugListener locationLine:<e.line> column:<e.pos>];"

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ObjC/ASTDbg.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

```

[The "BSD license"]

```

```

Copyright (c) 2006, 2007 Kay Roepke 2010 Alan Condit

```

```

All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright

notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
@genericParserHeaderFile.memVars() ::= <<
/* AST parserHeaderFile.memVars */
NSInteger ruleLevel;
NSArray *ruleNames;
<@super.memVars()> /* AST super.memVars */
<parserMemVars()> /* AST parserMemVars */
>>
```

```
@genericParserHeaderFile.properties() ::= <<
/* AST parserHeaderFile.properties */
<@super.properties()> /* AST super.properties */
<parserProperties()> /* AST parserproperties */
>>
```

```
@genericParserHeaderFile.methodsDecl() ::= <<
/* AST parserHeaderFile.methodsDecl */
<@super.methodsDecl()> /* AST super.methodsDecl */
<parserMethodsDecl()> /* AST parsermethodsDecl */
>>
```

```
@genericParser.synthesize() ::= <<
/* AST genericParser.synthesize */
<@super.synthesize()>
<parserSynthesize()>
>>
```

```
@genericParser.methods() ::= <<
/* AST genericParser.methods */
<@super.methods()>
<parserMethods()>
>>
```

```

/* additional init code for tree support */
@genericParser.init() ::= <<
/* AST genericParser.init */
<@super.init(>
[self setTreeAdaptor:[[CommonTreeAdaptor newTreeAdaptor] retain]];
>>

@genericParser.dealloc() ::= <<
/* AST genericParser.dealloc */
[self setTreeAdaptor:nil];
<@super.dealloc(>
>>

/* Add an adaptor property that knows how to build trees */
parserMemVars() ::= <<
/* AST parserMemVars */
id<TreeAdaptor> treeAdaptor;
>>

/* Add an adaptor property that knows how to build trees */
parserProperties() ::= <<
/* AST parserProperties */
@property (retain, getter=getTreeAdaptor, setter=setTreeAdaptor:) id<TreeAdaptor> treeAdaptor;
>>

/** Declaration of additional tree support methods - go in interface of parserHeaderFile() */
parserMethodsDecl() ::= <<
/* AST parserMethodsDecl */
- (id<TreeAdaptor>) getTreeAdaptor;
- (void) setTreeAdaptor:(id<TreeAdaptor>)theTreeAdaptor;
>>

/* Add an adaptor property that knows how to build trees */
parserSynthesize() ::= <<
/* AST parserProperties */
@synthesize treeAdaptor;
>>

/** Definition of addition tree support methods - go in implementation of genericParser() */
parserMethods() ::= <<
/* AST parserMethods */
- (id<TreeAdaptor>) getTreeAdaptor
{
return treeAdaptor;
}

- (void) setTreeAdaptor:(id<TreeAdaptor>)aTreeAdaptor
{

```

```

if (aTreeAdaptor != treeAdaptor) {
    treeAdaptor = aTreeAdaptor;
}
}
>>

/** addition memVars for returnscopes */
@returnScopeInterface.memVars() ::= <<
/* AST returnScopeInterface.memVars */
<recognizer.ASTLabelType; null="CommonTree"> *tree;
>>

/** the interface of returnScope properties */
@returnScopeInterface.properties() ::= <<
/* AST returnScopeInterface.properties */
@property (retain, getter=getTree, setter=setTree;) <recognizer.ASTLabelType; null="CommonTree"> *tree;
>>

/** the interface of returnScope methodsDecl */
@returnScopeInterface.methodsDecl() ::= <<
/* AST returnScopeInterface.methodsDecl */
- (<recognizer.ASTLabelType; null="CommonTree"> *)getTree;<\n>
- (void) setTree:(<recognizer.ASTLabelType; null="CommonTree"> *)aTree;<\n>
>>

/** the implementation of returnScope synthesize */
@returnScopeImplementation.synthesize() ::= <<
/* AST returnScope.synthesize */
@synthesize tree;
>>

/** the implementation of returnScope methods */
@returnScopeImplementation.methods() ::= <<
/* AST returnScope.methods */
- (<ASTLabelType> *)getTree
{
    return tree;
}

- (void) setTree:(<ASTLabelType> *)aTree
{
    if (tree != aTree) {
        if (tree != nil) [tree release];
        if (aTree != nil) [aTree retain];
        tree = aTree;
    }
}
}

```

```

- (void) dealloc
{
    self.tree = nil;
    [super dealloc];
}

>>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
/* AST ruleDeclarations */
<super.ruleDeclarations()>
<ASTLabelType> *root_0 = nil;<\n>
>>

ruleLabelDefs() ::= <<
/* AST ruleLabelDefs */
<super.ruleLabelDefs()>
<[ruleDescriptor.tokenLabels,ruleDescriptor.wildcardTreeLabels,
    ruleDescriptor.wildcardTreeListLabels]:{it | <ASTLabelType> *<it.label.text>_tree=nil;}; separator="\n">
<ruleDescriptor.tokenListLabels:{it | <ASTLabelType> *<it.label.text>_tree = nil;}; separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites:{it | RewriteRuleTokenStream *stream_<it> =
    [[RewriteRule<rewriteElementType>Stream newRewriteRule<rewriteElementType>Stream:treeAdaptor
        description:@"token <it>"] retain];}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites:{it | RewriteRuleSubtreeStream *stream_<it> =
    [[RewriteRuleSubtreeStream newRewriteRuleSubtreeStream:treeAdaptor
        description:@"rule <it>"] retain];}; separator="\n">
>>

ruleCleanUp() ::= <<
/* AST ruleCleanUp */
<super.ruleCleanUp()>
<[ruleDescriptor.allTokenRefsInAltsWithRewrites,ruleDescriptor.allRuleRefsInAltsWithRewrites]:{it |
    [stream_<it> release];}; separator="\n">
<!
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(backtracking)>if ( state.backtracking == 0 ) {<\n>
<endif>
    [<prevRuleRootRef()> setTree:(<ASTLabelType> *)[treeAdaptor rulePostProcessing:root_0]];<\n>
    [treeAdaptor setTokenBoundaries:[<prevRuleRootRef()> getTree]
        From:[<prevRuleRootRef()> getStart]
        To:[<prevRuleRootRef()> getStop]];<\n>
<if(backtracking)>}<\n>
<endif>
<endif>
[<root_0 release>];
!>
>>

```

```

rewriteCodeLabelsCleanup() ::= <<
/* AST rewriteCodeLabelsCleanup */
<referencedTokenLabels:{it | [stream_<it> release];}; separator="\n">
<referencedTokenListLabels:{it | [stream_<it> release];}; separator="\n">
<referencedRuleLabels:{it | [stream_<it> release];}; separator="\n">
<referencedRuleListLabels:{it | [stream_<it> release];}; separator="\n">
>>

/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
root_0 = (<ASTLabelType> *)[[[treeAdaptor class] newEmptyTree] retain];<\n>
<endif>
<endif>
<endif>
>>

// T r a c k i n g   R u l e   E l e m e n t s

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex) ::= <<
<! <super.tokenRef(...)> !>
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) <endif>
[stream_<token> addElement:<label>];<\n>
>>

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
 * to the tracking list stream_ID for use in the rewrite.
 */
tokenRefTrackAndListLabel(token,label,elementIndex) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex) ::= <<
<! <super.tokenRef(...)> !>
<tokenRefBang(...)>
<if(backtracking)>

```

```

if ( !<actions.(actionScope).synpredgate> ) <endif>
  [stream_<token> addElement:<label>];<\n>
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label,...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>
[stream_<rule.name> addElement:[<label> getTree]];
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabel(elem={ [<label> getTree] },...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<! <super.ruleRefRuleRoot(...)> !>
<ruleRefRuleRoot(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>
  [stream_<rule.name> addElement:[<label> getTree]];<\n>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabel(elem={ [<label> getTree] },...)>
>>

// R e w r i t e

rewriteCode(
  alts, description,
  referencedElementsDeep, // ALL referenced elements to right of ->
  referencedTokenLabels,
  referencedTokenListLabels,
  referencedRuleLabels,
  referencedRuleListLabels,
  referencedWildcardLabels,
  referencedWildcardListLabels,

```

```

rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::=
<<

// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
// wildcard labels: <[referencedWildcardLabels,referencedWildcardListLabels]; separator=", ">
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {<\n>
<endif>
<prevRuleRootRef(>.tree = root_0;<\n>
<rewriteCodeLabels(>
root_0 = (<ASTLabelType> *)[[[treeAdaptor class] newEmptyTree] retain];<\n>
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER)>
<if(rewriteMode)>
<prevRuleRootRef(>.tree = (<ASTLabelType>)[treeAdaptor rulePostProcessing:root_0];
[input replaceChildren:[treeAdaptor getParent:retval.start]
      From:[treeAdaptor getChildIndex:retval.start]
      To:[treeAdaptor getChildIndex:_last]
      With:retval.tree];
<endif>
<endif>
<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER)>
<prevRuleRootRef(>.tree = root_0;<\n>
<else>
<if(!rewriteMode)>
<prevRuleRootRef(>.tree = root_0;<\n>
<endif>
<endif>
<if(backtracking)>
}
<endif>
>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
: {it | RewriteRule<rewriteElementType>Stream *stream_<it> =
  [[RewriteRule<rewriteElementType>Stream newRewriteRule<rewriteElementType>Stream:treeAdaptor
description:@"token <it>" element:<it>] retain];};
  separator="\n"
>
<referencedTokenListLabels: {it | RewriteRule<rewriteElementType>Stream *stream_<it> =

```

```

[[RewriteRule<rewriteElementType>Stream newRewriteRule<rewriteElementType>Stream:treeAdaptor
  description:@ "token <it>" elements:list_<it>] retain];};
separator="\n"
>
<referencedWildcardLabels:{it | RewriteRuleSubtreeStream stream_<it> =
  [[RewriteRuleSubtreeStream newRewriteRuleSubtreeStream:treeAdaptor
    description:"wildcard <it>" element:<it>] retain];};
separator="\n"
>
<referencedWildcardListLabels:{it | RewriteRuleSubtreeStream stream_<it> =
  [[RewriteRuleSubtreeStream newRewriteRuleSubtreeStream:treeAdaptor
    descriptor:"wildcard <it>" elements:list_<it>] retain];};
separator="\n"
>
<referencedRuleLabels:{it | RewriteRuleSubtreeStream *stream_<it> =
  [[RewriteRuleSubtreeStream newRewriteRuleSubtreeStream:treeAdaptor
    description:@ "token <it>" element:<it>!=nil?[<it> getTree]:nil] retain];};
separator="\n"
>
<referencedRuleListLabels:{it | RewriteRuleSubtreeStream *stream_<it> =
  [[RewriteRuleSubtreeStream newRewriteRuleSubtreeStream:treeAdaptor
    description:@ "token <it>" elements:list_<it>] retain];};
separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
if ( <referencedElementsDeep:{el | [stream_<el> hasNext]}; separator="||"> ) {
  <alt>
}
<referencedElementsDeep:{el | [stream_<el> reset];<n>}>
>>

rewriteClosureBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<

```



```

// <fileName>:<description>
while ( <referencedElements:{el | [stream_<el> hasNext]}; separator="|"> ) {
  <alt>
}
<referencedElements:{el | [stream_<el> reset];<\n>}>
>>

rewritePositiveClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
// <fileName>:<description>
{
if ( !(<referencedElements:{el | [stream_<el> hasNext]}; separator="|">) ) {
  @throw [RewriteEarlyExitException newException];
}
while ( <referencedElements:{el | [stream_<el> hasNext]}; separator="|"> ) {
  <alt>
}
<referencedElements:{el | [stream_<el> reset];<\n>}>
}
>>

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>) {
  <a.alt>
}<\n>
<else>
{
  <a.alt>
}<\n>
<endif>
>>

/** For empty rewrites: "r: ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = nil;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
  <ASTLabelType> *root_<treeLevel> = (<ASTLabelType> *)[[[treeAdaptor class] newEmptyTree] retain];
  <root:rewriteElement()>
  <children:rewriteElement()>
  [treeAdaptor addChild:root_<treeLevel> toTree:root_<enclosingTreeLevel>];
}

```

```

}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,terminalOptions,args) ::= <<
// TODO: args: <args; separator=", ">
[treeAdaptor addChild:<createRewriteNodeFromElement(...)> toTree:root_<treeLevel>];<\n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
[treeAdaptor addChild:[stream_<label> nextNode] toTree:root_<treeLevel>];<\n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
[treeAdaptor addChild:[stream_<label> nextNode] toTree:root_<treeLevel>];<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:[stream_<label> nextNode]
old:root_<treeLevel>];<\n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,terminalOptions,args) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:<createRewriteNodeFromElement(...)>
old:root_<treeLevel>];<\n>
>>

rewriteImaginaryTokenRef(args,token,terminalOptions,elementIndex) ::= <<
[treeAdaptor addChild:<createImaginaryNode(tokenType=token, ...)> toTree:root_<treeLevel>];<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,terminalOptions,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:<createImaginaryNode(tokenType=token, ...)>
old:root_<treeLevel>];<\n>

```

```

>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
 * let's us refer to $rule to mean previous value. I am reusing the
 * variable 'tree' sitting in retval struct to hold the value of root_0 right
 * before I set it during rewrites. The assign will be to retval.tree.
 */
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
[treeAdaptor addChild:[stream_<rule> nextTree] toTree:root_<treeLevel>];<\n>
>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:(id<Tree>)[stream_<rule> nextNode]
old:root_<treeLevel>];<\n>
>>

rewriteNodeAction(action) ::= <<
[treeAdaptor addChild:<action> toTree:root_<treeLevel>];<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:<action> old:root_<treeLevel>];<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
[treeAdaptor addChild:[stream_<label> nextTree] toTree:root_<treeLevel>];<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
[treeAdaptor addChild:[stream_<label> nextTree] toTree:root_<treeLevel>];<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:[stream_<label> nextNode]
old:root_<treeLevel>];<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */

```

```

rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:[stream_<label> nextNode]
old:root_<treeLevel>];<n>
>>

rewriteWildcardLabelRef(label) ::= <<
[treeAdaptor addChild:[stream_<label> nextTree] toTree:root_<treeLevel>];<n>
>>

createImaginaryNode(tokenType,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
  [<terminalOptions.node> new<terminalOptions.node>:<tokenType> <if(args)>, <args; separator=", "><endif>]
<else>
  <if(args)>
    [[treeAdaptor createTree:<tokenType> <if(first(args))>FromToken:<first(args)><endif>
<if(first(rest(args)))>Text:<first(rest(args))><else>Text:@ "<tokenType>"<endif>] retain]
  <else>
    [[treeAdaptor createTree:<tokenType> Text:@ "<tokenType>"] retain]
  <endif>
<endif>
>>

```

```

createRewriteNodeFromElement(token,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
  [<terminalOptions.node> new<terminalOptions.node>:[stream_<token> nextToken]<if(args)>, <args;
separator=", "><endif>]
<else>
  <if(args)> <! must create new node from old !>
    [[treeAdaptor createTree:<token> Text:<first(rest(args))> <args; separator=", ">] retain]
  <else>
    [stream_<token> nextNode]
  <endif>
<endif>
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/ObjC/AST.stg
No license file was found, but licenses were detected in source scan.

```

/*
* [The "BSD license"]
* Copyright (c) 2011 Terence Parr
* All rights reserved.
*
* Conversion to C#:
* Copyright (c) 2011 Sam Harwell, Pixel Mine, Inc.
* All rights reserved.
*

```

```

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

@outputFile.imports() ::= <<
<@super.imports()>

```

```

<if(!TREE_PARSER)>
<! tree parser would already have imported !>
using Antlr.Runtime.Tree;
using RewriteRuleITokenStream = Antlr.Runtime.Tree.RewriteRuleTokenStream;
<endif>
>>

```

```

@genericParser.members() ::= <<
<@super.members()>
<parserMembers()>
>>

```

```

parserCtorBody() ::= <<
<super.parserCtorBody()>
<treeAdaptorType()> treeAdaptor = default(<treeAdaptorType()>);
CreateTreeAdaptor(ref treeAdaptor);
TreeAdaptor = treeAdaptor<if(!actions.(actionScope).treeAdaptorType)> ?? new CommonTreeAdaptor()<endif>;
>>

```

```

/** Add an adaptor property that knows how to build trees */
parserMembers() ::= <<
// Implement this function in your helper file to use a custom tree adaptor

```

```

partial void CreateTreeAdaptor(ref <treeAdaptorType()> adaptor);

private <treeAdaptorType()> adaptor;

public <treeAdaptorType()> TreeAdaptor
{
    get
    {
        return adaptor;
    }

    set
    {
        this.adaptor = value;
        <grammar.directDelegates:{g|<g:delegateName()>.TreeAdaptor = this.adaptor;}>
    }
}
>>

treeAdaptorType() ::= <<
<actions.(actionScope).treeAdaptorType; null="ITreeAdaptor">
>>

ruleReturnBaseType() ::= <%
Ast<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope\<<ASTLabelType>, <labelType>>
%>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
<ASTLabelType> root_0 = default(<ASTLabelType>);<\n>
>>

ruleLabelDefs(ruleDescriptor, labelType, ASTLabelType, rewriteElementType) ::= <%
<super.ruleLabelDefs(...)>
<if(!ruleDescriptor.isSynPred)>
<[ruleDescriptor.tokenLabels,ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it|\n><ASTLabelType> <it.label.text>_tree = default(<ASTLabelType>);}>
<ruleDescriptor.tokenListLabels: {it|\n><ASTLabelType> <it.label.text>_tree = default(<ASTLabelType>);}>
<ruleDescriptor.allTokenRefsInAltsWithRewrites
: {it|\n>RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>");}>
<ruleDescriptor.allRuleRefsInAltsWithRewrites
: {it|\n>RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule <it>");}>
<endif>
%>

/** When doing auto AST construction, we must define some variables;

```

```

* These should be turned off if doing rewrites. This must be a "mode"
* as a rule could have both rewrite and AST within the same alternative
* block.
*/
@alt.declarations() ::= <<
<if(autoAST && outerAlt && !rewriteMode && !ruleDescriptor.isSynPred)>
root_0 = (<ASTLabelType>)adaptor.Nil();
<endif>
>>

// Tracking Rule Elements

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<token>.Add(<label>);<\n>
>>

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
* to the tracking list stream_ID for use in the rewrite.
*/
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefTrack(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<token>.Add(<label>);
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<rule.name>.Add(<label>.Tree);
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabelElem(elem={ <label>.Tree },elemType=ASTLabelType,...)>

```

```

>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>stream_<rule>.Add(<label>.Tree);
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

// R e w r i t e

rewriteCode(
alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
referencedWildcardLabels,
referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::= <<
<\n>{
// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
// wildcard labels: <[referencedWildcardLabels,referencedWildcardListLabels]; separator=", ">
<if(backtracking)>
if (<actions.(actionScope).synpredgate>) {
<endif>
<prevRuleRootRef().Tree = root_0;
<rewriteCodeLabels()>
root_0 = (<ASTLabelType>)adaptor.Nil();
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER&&rewriteMode)>
<prevRuleRootRef().Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
if (<prevRuleRootRef().Tree != null)
input.ReplaceChildren(adaptor.GetParent(retval.Start), adaptor.GetChildIndex(retval.Start),
adaptor.GetChildIndex(_last), retval.Tree);
<endif>

```



```

<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER||!rewriteMode)>
<prevRuleRootRef().Tree = root_0;
<endif>
<if(backtracking)>
}
<endif>
}

>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
:{it|RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>",<it>)};
separator="\n"
>
<referencedTokenListLabels
:{it|RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it> ", list_<it>)};
separator="\n"
>
<referencedWildcardLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedWildcardListLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it> ",list_<it>)};
separator="\n"
>
<referencedRuleLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule
<it>",<it>!=null?<it>.Tree:null)};
separator="\n"
>
<referencedRuleListLabels
:{it|RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"token <it> ",list_<it>)};
separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks

```

```

description) ::=
<<
// <fileName>:<description>
if (<referencedElementsDeep:{el | stream_<el>.HasNext}; separator="||">)
{
  <alt>
}
<referencedElementsDeep:{el | stream_<el>.Reset();<\n>}>
>>

```

```

rewriteClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
// <fileName>:<description>
while ( <referencedElements:{el | stream_<el>.HasNext}; separator="||"> )
{
  <alt>
}
<referencedElements:{el | stream_<el>.Reset();<\n>}>
>>

```

```

rewritePositiveClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
if (!(<referencedElements:{el | stream_<el>.HasNext}; separator="||">))
{
  throw new RewriteEarlyExitException();
}
while ( <referencedElements:{el | stream_<el>.HasNext}; separator="||"> )
{
  <alt>
}
<referencedElements:{el | stream_<el>.Reset();<\n>}>
>>

```

```

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>)
{
  <a.alt>
}

```

```

<else>
{
  <a.alt>
}
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
  <ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.Nil();
  <root:rewriteElement()>
  <children:rewriteElement()>
  adaptor.AddChild(root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <%
<@pregen()>
DebugLocation(<e.line>, <e.pos>);<\n>
<e.el>
%>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,args,terminalOptions={}) ::= <<
adaptor.AddChild(root_<treeLevel>, <createRewriteNodeFromElement(...)>);<\n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextNode());<\n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextNode());<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

```

```

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,args,terminalOptions={}) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<createRewriteNodeFromElement(...)>,
root_<treeLevel>);<\n>
>>

rewriteImaginaryTokenRef(args,token,elementIndex,terminalOptions={}) ::= <<
adaptor.AddChild(root_<treeLevel>, <createImaginaryNode(tokenType=token, ...)>);<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,elementIndex,terminalOptions={}) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<createImaginaryNode(tokenType=token, ...)>,
root_<treeLevel>);<\n>
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<rule>.NextTree());<\n>
>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<rule>.NextNode(), root_<treeLevel>);<\n>
>>

rewriteNodeAction(action) ::= <<
adaptor.AddChild(root_<treeLevel>, <action>);<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<action>, root_<treeLevel>);<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<

```

```

adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(stream_<label>.NextNode(), root_<treeLevel>);<\n>
>>

rewriteWildcardLabelRef(label) ::= <<
adaptor.AddChild(root_<treeLevel>, stream_<label>.NextTree());<\n>
>>

createImaginaryNode(tokenType,args,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
<! new MethodNode(IDLabel, args) !>
new <terminalOptions.node><(tokenType)<if(args)>, <args; separator=", "><endif>
<else>
(<ASTLabelType>)adaptor.Create(<tokenType>, <args; separator=", "><if(!args)>"<tokenType>"<endif>
<endif>
%>

createRewriteNodeFromElement(token,args,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.NextToken())<if(args)>, <args; separator=", "><endif>
<else>
<if(args)> <! must create new node from old !>
adaptor.Create(<token>, <args; separator=", ">)
<else>
stream_<token>.NextNode()
<endif>
<endif>
%>

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/AST.stg

```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Template subgroup to add template rewrite output
```

```
* If debugging, then you'll also get STDbg.stg loaded.
```

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
import stringtemplate3
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturnInit() ::= <<
```

```
self.st = None
```

```
>>
```

```
@returnScope.ruleReturnMembers() ::= <<
```

```
def getTemplate(self):
```

```
    return self.st
```

```
def toString(self):
```

```
    if self.st is not None:
```

```
        return self.st.toString()
```

```

    return None
    __str__ = toString

>>

@genericParser.init() ::= <<
<@super.init()>
self.templateLib = stringtemplate3.StringTemplateGroup(
    '<name>Templates', lexer='angle-bracket'
)

>>

@genericParser.members() ::= <<
<@super.members()>
def setTemplateLib(self, templateLib):
    self.templateLib = templateLib

def getTemplateLib(self):
    return self.templateLib

>>

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.st})>
>>

rewriteTemplate(alts) ::= <<
# TEMPLATE REWRITE
<if(backtracking)>
if <actions.(actionScope).synpredgate>:
    <first(alts):rewriteTemplateAltFirst()>
    <rest(alts):{it | el<rewriteTemplateAlt(it)>}>
    <if(rewriteMode)><replaceTextInLine()><endif>

<else>
<first(alts):rewriteTemplateAltFirst()>
<rest(alts):{it | el<rewriteTemplateAlt(it)>}>
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>

replaceTextInLine() ::= <<
<if(TREE_PARSER)>
self.input.getTokenStream().replace(
    self.input.getTreeAdaptor().getTokenStartIndex(retval.start),

```

```

        self.input.getTreeAdaptor().getTokenStopIndex(retval.start),
        retval.st
    )
<else>
self.input.replace(
    retval.start.getTokenIndex(),
    self.input.LT(-1).getTokenIndex(),
    retval.st
)
<endif>
>>

```

```

rewriteTemplateAltFirst(alt) ::= <<
<if(alt.pred)>
if <alt.pred>:
    # <alt.description>
    retval.st = <alt.alt>
<\n>
<else>
# <alt.description>
retval.st = <alt.alt>
<\n>
<endif>
>>

```

```

rewriteTemplateAlt(alt) ::= <<
<if(alt.pred)>if <alt.pred>:
    # <alt.description>
    retval.st = <alt.alt>
<\n>
<else>se:
    # <alt.description>
    retval.st = <alt.alt>
<\n>
<endif>
>>

```

```

rewriteEmptyTemplate(alts) ::= <<
None
>>

```

```

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template *after* having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded
 * template.
 */

```

```

rewriteExternalTemplate(name,args) ::= <%

```



```

self.templateLib.getInstanceOf("<name><if(args)>, attributes={<args:{a | "<a.name>": <a.value>}; separator=",
"><endif>)
%>

/** expr is a string expression that says what template to load */
rewriteIndirectTemplate(expr,args) ::= <%
self.templateLib.getInstanceOf(<expr><if(args)>, attributes={<args:{a | "<a.name>": <a.value>}; separator=",
"><endif>)
%>

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <%
stringtemplate3.StringTemplate("<template>", group=self.templateLib<if(args)>, attributes={<args:{a |
"<a.name>": <a.value>}; separator=", "><endif>)
%>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
<action>
>>

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
(<st>["<attrName>"] = <expr>
>>

/** Translate %{stringExpr} */
actionStringConstructor(stringExpr) ::= <<
stringtemplate3.StringTemplate(<stringExpr>, group=self.templateLib)
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Python/ST.stg

No license file was found, but licenses were detected in source scan.

```

/*
* [The "BSD license"]
* Copyright (c) 2007-2008 Johannes Luber
* Copyright (c) 2005-2007 Kunle Odutola
* Copyright (c) 2011 Sam Harwell
* Copyright (c) 2011 Terence Parr
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.

```

```

* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** Template overrides to add debugging to normal Java output;
* If ASTs are built, then you'll also get ASTDbg.stg loaded.
*/

```

```

@outputFile.debugPreprocessor() ::= "#define ANTLR_DEBUG"

```

```

@outputFile.imports() ::= <<
<@super.imports(>
using Antlr.Runtime.Debug;
using IOException = System.IO.IOException;
>>

```

```

@genericParser.members() ::= <<
<if(grammar.grammarIsRoot)>
public static readonly string[] ruleNames =
new string[]
{
    "invalidRule", <grammar.allImportedRules:{rST | "<rST.name>"}; wrap="\n ", separator=", ">
};<\n>
<endif>
<if(grammar.grammarIsRoot)><! grammar imports other grammar(s) !>
int ruleLevel = 0;
public virtual int RuleLevel { get { return ruleLevel; } }
public virtual void IncRuleLevel() { ruleLevel++; }
public virtual void DecRuleLevel() { ruleLevel--; }
<if(profile)>
<ctorForProfilingRootGrammar(>
<else>
<ctorForRootGrammar(>
<endif>
<ctorForPredefinedListener(>

```

```

<else><! imported grammar !>
public int RuleLevel { get { return <grammar.delegators:{g| <g:delegateName()>>>.RuleLevel; } }
public void IncRuleLevel() { <grammar.delegators:{g| <g:delegateName()>>>.IncRuleLevel(); }
public void DecRuleLevel() { <grammar.delegators:{g| <g:delegateName()>>>.DecRuleLevel(); }
<ctorForDelegateGrammar()>
<endif>
<if(profile)>
public override bool AlreadyParsedRule( IIntStream input, int ruleIndex )
{
int stopIndex = GetRuleMemoization(ruleIndex, input.Index);
((Profiler)dbg).ExamineRuleMemoization(input, ruleIndex, stopIndex,
<grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
return base.AlreadyParsedRule(input, ruleIndex);
}<\n>
public override void Memoize( IIntStream input, int ruleIndex, int ruleStartIndex )
{
((Profiler)dbg).Memoize(input, ruleIndex, ruleStartIndex,
<grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
base.Memoize(input, ruleIndex, ruleStartIndex);
}<\n>
<endif>
protected virtual bool EvalPredicate( bool result, string predicate )
{
dbg.SemanticPredicate( result, predicate );
return result;
}<\n>
>>

ctorForRootGrammar() ::= <<
<! bug: can't use <@super.members()> cut-n-paste instead !>
<! Same except we add port number and profile stuff if root grammar !>
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input )
: this( input, DebugEventSocketProxy.DefaultDebuggerPort, new RecognizerSharedState() )
{
}
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, int port,
RecognizerSharedState state )
: base( input, state )
{
<createListenerAndHandshake()>
<grammar.directDelegates:{g|<g:delegateName()> = new <g.recognizerName>( input, dbg, this.state,
this<grammar.delegators:{g|, <g:delegateName()>>> );}; separator="\n">
<parserCtorBody()>
<@finally()>
}<\n>
>>

ctorForProfilingRootGrammar() ::= <<

```

```

<! bug: can't use <@super.members()> cut-n-paste instead !>
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input )
: this( input, new Profiler(null), new RecognizerSharedState() )
{
}
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg,
RecognizerSharedState state )
: base( input, dbg, state )
{
Profiler p = (Profiler)dbg;
p.setParser(this);
<grammar.directDelegates:
{g|<g:delegateName()> = new <g.recognizerName>( input, dbg, this.state, this<grammar.delegators:{g|,
<g:delegateName()>>> );}; separator="\n">
<parserCtorBody()>
<@finally()>
}
}<\n>
>>

/** Basically we don't want to set any dbg listeners are root will have it. */
ctorForDelegateGrammar() ::= <<
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg,
RecognizerSharedState state<grammar.delegators:{g|, <g.recognizerName> <g:delegateName()>>> )
: base( input, dbg, state )
{
<grammar.directDelegates:
{g|<g:delegateName()> = new <g.recognizerName>( input, this, this.state<grammar.delegators:{g|,
<g:delegateName()>>> );}; separator="\n">
<parserCtorBody()>
}<\n>
}>>

ctorForPredefinedListener() ::= <<
<actions.(actionScope).ctorModifier; null="public"> <name>( <inputStreamType> input, IDebugEventListener dbg
)
<@superClassRef>: base( input, dbg, new RecognizerSharedState() )<@end>
{
<if(profile)>
Profiler p = (Profiler)dbg;
p.setParser(this);
<endif>
<grammar.directDelegates:{g|<g:delegateName()> = new <g.recognizerName>(input, dbg, this.state,
this<grammar.delegators:{g|, <g:delegateName()>>>);}; separator="\n">
<parserCtorBody()>
<@finally()>
}<\n>
}>>

```

```

createListenerAndHandshake() ::= <<
<if(TREE_PARSER)>
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port, input.TreeAdaptor );<\n>
<else>
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port, null );<\n>
<endif>
DebugListener = proxy;
try
{
proxy.Handshake();
}
catch ( IOException ioe )
{
ReportError( ioe );
}
>>

@genericParser.superClassName() ::= "Debug<@super.superClassName()>"

/*
* Many of the following rules were merged into CSharp2.stg.
*/

@rule.preamble() ::= <<
if (RuleLevel == 0)
DebugListener.Commence();
IncRuleLevel();
>>
//@rule.preamble() ::= <<
//try
//{
// dbg.EnterRule( GrammarFileName, "<ruleName>" );
// if ( RuleLevel == 0 )
// {
// dbg.Commence();
// }
// IncRuleLevel();
// dbg.Location( <ruleDescriptor.tree.line>, <ruleDescriptor.tree.charPositionInLine> );<\n>
//>>

@rule.postamble() ::= <<
DecRuleLevel();
if (RuleLevel == 0)
DebugListener.Terminate();
>>
//@rule.postamble() ::= <<
//dbg.Location(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.charPositionInLine>);<\n>

```

```

//}
//finally
//{
// dbg.ExitRule( GrammarFileName, "<ruleName>" );
// DecRuleLevel();
// if ( RuleLevel == 0 )
// {
// dbg.Terminate();
// }
//}<\n>
//>>

@@insertSynpreds.start() ::= "dbg.BeginBacktrack( state.backtracking );"
@@insertSynpreds.stop() ::= "dbg.EndBacktrack( state.backtracking, success );"

// Common debug event triggers used by region overrides below

//enterSubRule() ::= <<
//try
//{
// dbg.EnterSubRule( <decisionNumber> );<\n>
//>>

//exitSubRule() ::= <<
//}
//finally
//{
// dbg.ExitSubRule( <decisionNumber> );
//}<\n>
//>>

//enterDecision() ::= <<
//try
//{
// dbg.EnterDecision( <decisionNumber> );<\n>
//>>

//exitDecision() ::= <<
//}
//finally
//{
// dbg.ExitDecision( <decisionNumber> );
//}<\n>
//>>

//enterAlt(n) ::= "dbg.EnterAlt( <n> );<\n>"

// Region overrides that tell various constructs to add debugging triggers

```

```

//@block.predecision() ::= "<enterSubRule()><enterDecision()>"

//@block.postdecision() ::= "<exitDecision()>"

//@block.postbranch() ::= "<exitSubRule()>"

//@ruleBlock.predecision() ::= "<enterDecision()>"

//@ruleBlock.postdecision() ::= "<exitDecision()>"

//@ruleBlockSingleAlt.preal() ::= "<enterAlt(n=\"1\")>"

//@blockSingleAlt.preal() ::= "<enterAlt(n=\"1\")>"

//@positiveClosureBlock.preloop() ::= "<enterSubRule()>"

//@positiveClosureBlock.postloop() ::= "<exitSubRule()>"

//@positiveClosureBlock.predecision() ::= "<enterDecision()>"

//@positiveClosureBlock.postdecision() ::= "<exitDecision()>"

//@positiveClosureBlock.earlyExitException() ::=
// "dbg.RecognitionException( eee<decisionNumber> );<n>"

//@closureBlock.preloop() ::= "<enterSubRule()>"

//@closureBlock.postloop() ::= "<exitSubRule()>"

//@closureBlock.predecision() ::= "<enterDecision()>"

//@closureBlock.postdecision() ::= "<exitDecision()>"

//@altSwitchCase.preal() ::= "<enterAlt(n=i)>"

//@element.prematch() ::=
// "dbg.Location( <it.line>, <it.pos> );"

//@matchSet.mismatchedSetException() ::=
// "dbg.RecognitionException( mse );"

//@dfaState.noViableAltException() ::= "dbg.RecognitionException( nvae );"

//@dfaStateSwitch.noViableAltException() ::= "dbg.RecognitionException( nvae );"

//dfaDecision(decisionNumber,description) ::= <<
//try

```

```

//{{
// isCyclicDecision = true;
// <super.dfaDecision(...)>
//}}
//catch ( NoViableAltException nvae )
//{{
// dbg.RecognitionException( nvae );
// throw nvae;
//}}
//>>

/**@cyclicDFA.errorMethod() ::= <<
//public override void Error( NoViableAltException nvae )
//{{
// ((DebugParser)recognizer).dbg.RecognitionException( nvae );
//}}
//>>

/** Force predicate validation to trigger an event */
evalPredicate(pred,description) ::= <<
EvalPredicate(<pred>, "<description>")
>>

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp2/Dbg.stg

```

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2007 Kay Roepke 2010 Alan Condit

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during normal parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* The situation is not too bad as rewrite (->) usage makes ^ and !

* invalid. There is no huge explosion of combinations.

*/

@rule.setErrorReturnValue() ::= <<

/* ASTParser rule.setErrorReturnValue */

retval.tree = (<ASTLabelType> *)[treeAdaptor errorNode:input From:retval.start To:[input LT:-1] Exception:re];

<! System.out.println("<ruleName> returns "+((CommonTree)retval.tree).toStringTree()); !>

>>

// TOKEN AST STUFF

/** ID and output=AST */

tokenRef(token, label, elementIndex, terminalOptions) ::= <<

/* ASTParser tokenRef */

<super.tokenRef(...)>

<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>

<label>_tree = <createNodeFromToken(...)>;

[treeAdaptor addChild:<label>_tree toTree:root_0];

<if(backtracking)>}<endif>

>>

/* ID! and output=AST (same as plain tokenRef) */

/* ASTParser tokenRefBang */

tokenRefBang(token,label,elementIndex,terminalOptions) ::= "<super.tokenRef(...)>"

/** ID^ and output=AST */

tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<

<super.tokenRef(...)>

<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>

<label>_tree = <createNodeFromToken(...)>;

root_0 = (<ASTLabelType> *)[treeAdaptor becomeRoot:<label>_tree old:root_0];

<if(backtracking)>}<endif>

```

>>

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
/* ASTParser tokenRefBangAndListLabel */
<tokenRefBang(...)>
<listLabel(elem=label,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
/* ASTParser tokenRefAndListLabel */
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */
tokenRefRuleRootAndListLabel(token,label,terminalOptions,elementIndex) ::= <<
/* ASTParser tokenRefRuleRootAndListLabel */
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <%
/* ASTParser matchSet */
<super.matchSet(postmatchCode={<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
 [treeAdaptor addChild:<createNodeFromToken(...)> toTree:root_0 ];}, ...)>
%>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
/* ASTParser matchRuleBlockSet */
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,terminalOptions, postmatchCode) ::= "<super.matchSet(...)>"

```

```

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
/* ASTParser matchSetRuleRoot */
<if(label)>
<label>=(<labelType> *)[input LT:1]; /* matchSetRuleRoot */<\n>
<endif>
<super.matchSet(postmatchCode={ <if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
root_0 = (<ASTLabelType> *)[treeAdaptor becomeRoot:<createNodeFromToken(...)> old:root_0];}, ...)>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
/* ASTParser ruleRef */
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
[treeAdaptor addChild:[<label> getTree] toTree:root_0];
>>

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
/* ASTParser ruleRefRuleRoot */
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
root_0 = (<ASTLabelType> *)[treeAdaptor becomeRoot:[<label> getTree] old:root_0];
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ASTParser ruleRefAndListLabel */
<ruleRef(...)>
<listLabel(elem = {[<label> getTree]},...)>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ASTParser ruleRefBangAndListLabel */
<ruleRefBang(...)>
<listLabel(elem = {[<label> getTree]},...)>
>>

```

```

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ASTParser ruleRefRuleRootAndListLabel */
<ruleRefRuleRoot(...)>
<listLabel(elem = {[<label> getTree]},...)>
>>

// WILDCARD AST

wildcard(token,label,elementIndex,terminalOptions) ::= <<
/* ASTParser wildcard */
<super.wildcard(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
  [treeAdaptor addChild:[treeAdaptor create:<label>] retain] toTree:root_0];
<if(backtracking)>}<endif>
>>

wildcardBang(token,label,elementIndex,terminalOptions) ::= "<super.wildcard(...)>"

wildcardRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
/* ASTParser wildcardRuleRoot */
<super.wildcard(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
  <label>_tree = [[treeAdaptor create:<label>] retain]
  root_0 = (<ASTLabelType> *)[treeAdaptor becomeRoot:<label>_tree old:root_0];
<if(backtracking)>}<endif>
>>

createNodeFromToken(label,terminalOptions) ::= <<
/* ASTParser createNodeFromToken */
<if(terminalOptions.node)>
[<terminalOptions.node> new<terminalOptions.node>:<label>] <! new MethodNode(IDLabel) !>
<else>
(<ASTLabelType> *)[treeAdaptor create:<label>] retain]
<endif>
>>

// straight from java cleanup ///
ruleCleanUp() ::= <<
/* ASTParser ruleCleanUp */
<super.ruleCleanUp()>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<\n><endif>
  retval.tree = (<ASTLabelType> *)[treeAdaptor rulePostProcessing:root_0];
  [treeAdaptor setTokenBoundaries:retval.tree From:retval.start To:retval.stopToken];
<if(backtracking)>}<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/ObjC/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Template subgroup to add template rewrite output
```

```
* If debugging, then you'll also get STDbg.stg loaded.
```

```
*/
```

```
@returnScopeInterface.memVars() ::= <<
```

```
<@super.memVars()>
```

```
/* ST returnInterface.memVars */
```

```
ST *st;
```

```
>>
```

```
@returnScopeInterface.properties() ::= <<
```

```
<@super.properties()>
```

```
/* ST returnScope.properties */
```

```
@property (retain, getter=getST, setter=setST:) ST *st;
```

```
>>
```

```
@returnScopeInterface.methodsDecl() ::= <<
```

```

<@super.methodsDecl()>
/* ST AST returnScopeInterface.methodsDecl */
- (id) getTemplate;
- (NSString *) toString;
- (NSString *) description;
>>

@returnScopeInterface() ::= <<
/* ST returnScopeInterface */
@interface <returnScopeInterface.name> : ReturnScope {
<returnScopeInterface.memVars()>
}
<returnScopeInterface.properties()>

<returnScopeInterface.methods()>
@end
>>

@returnScopeImplementation.synthesize() ::= <<
<@super.synthesize()>
/* ST returnScope.synthesize */
@synthesize st;
>>

@returnScopeImplementation.methods() ::= <<
<@super.methods()>
/* ST returnScope.methods */
- (id) getTemplate { return st; }
- (NSString *) toString { return st==nil?nil:[st toString]; }
- (NSString *) description { [self toString]; }
>>

@returnScopeImplementation() ::= <<
@implementation <returnScopeImplementation.name>
<returnScopeImplementation.synthesize()>

<returnScopeImplementation.methods()>
@end
>>

/** Add this to each rule's return value struct */
@returnScope.ruleReturnMembers() ::= <<
<@super.ruleReturnMembers()>
/* ST returnScope.ruleReturnMembers -- empty */
>>

@genericParserHeaderFile.memVars() ::= <<
<@super.memVars()>

```

```

/* ST genericParserHeaderFile.memVars -- empty now */
STGroup *templateLib; /* ST -- really a part of STAttrMap */
>>

@genericParserHeaderFile.properties() ::= <<
<@super.properties()>
/* ST genericParser.properties */
@property (retain, getter=getTemplateLib, setter=setTemplateLib:) STGroup *templateLib;
>>

@genericParserHeaderFile.methodsDecl() ::= <<
<@super.methodsDecl()>
/* ST genericParser.methodsDecl */
- init;
- (STGroup *) getTemplateLib;
- (void) setTemplateLib:(STGroup *)aTemplateLib;
@end
>>

@genericParser.synthesize() ::= <<
<@super.synthesize()>
/* ST genericParserImplementation.synthesize */
@synthesize templateLib;
>>

@genericParser.methods() ::= <<
<@super.methods()>
/* ST genericParser.methods */

- (STGroup *)getTemplateLib
{
    return templateLib;
}

- (void) setTemplateLib:(STGroup *)aTemplateLib
{
    templateLib = aTemplateLib;
}

>>

@genericParser.members() ::= <<
<@super.members()>
STGroup *templateLib = [STGroup newSTGroup:@"<name>Templates"];

- (STGroup *) getTemplateLib
{
    return templateLib;
}

```

```

}

- (void) setTemplateLib:(STGroup *) templateLib
{
    this.templateLib = templateLib;
}

/** allows convenient multi-value initialization:
 * "new STAttrMap().put(...).put(...)"
 */
/* REPLACE THIS STATIC CLASS
static class STAttrMap extends HashMap {
- (STAttrMap *) setObject:(id)aValue forKey:(NS*)String attrName
{
    [super setObject:value forKey:attrName];
    return self;
}
- (STAttrMap *) setObjectWithInt:(NSInteger)value forKey:(NSString *)attrName
{
    [super setObject:[NSNumber numberWithInt:value] forKey:attrName];
    return self;
}
}
*/
>>

@STAttrMap() ::= <<
/* ----- ST start STAttrMap ----- */
<@STAttrMap.interface()>
<@STAttrMap.implementation()>
/* ----- ST end STAttrMap ----- */
>>

@STAttrMap.interface() ::= <<
/* ----- ST start STAttrMap.interface ----- */
@interface STAttrMap : HashMap {
/* <@STAttrMap.memVars()> */
    STGroup *templateLib;
}

/* <@STAttrMap.properties()> */
@property (retain, getter=getTemplateLib, setter=setTemplateLib:) STGroup *templateLib;
/* <@STAttrMap.methodsDecl()> */
- (id) init;
- (STAttrMap *) setObject:(id)value forKey:(NSString *)attrName;
- (STAttrMap *) setObjectWithInt:(NSInteger)value forKey:(NSString *)attrName;
- (void) setTemplateLib:(STGroup *)aTemplateLib;
- (STGroup *) getTemplateLib;

```



```

@end
/* ----- ST end STAttrMap.interface ----- */
>>

@STAttrMap.implementation() ::= <<
/* ----- ST start STAttrMap.implementation ----- */
/** allows convenient multi-value initialization:
 * "new STAttrMap().put(...).put(...)"
 */
@implementation STAttrMap
@synthesize templateLib;

<@STAttrMap.methods()>
@end
/* ----- ST end STAttrMap.implementation ----- */
>>

@STAttrMap.memVars() ::= <<
/* ----- ST start STAttrMap.memVars ----- */
    STGroup *templateLib;
/* ----- ST end STAttrMap.memVars ----- */
>>

@STAttrMap.properties() ::= <<
/* ----- ST start STAttrMap.properties ----- */
@property (retain, getter=getTemplateLib, setter=setTemplateLib:) STGroup *templateLib;
/* ----- ST end STAttrMap.properties ----- */
>>

@STAttrMap.methodsDecl() ::= <<
/* ----- ST start STAttrMap.methodsDecl ----- */
- (id) init;
- (STAttrMap *) setObject:(id)value forKey:(NSString *)attrName;
- (STAttrMap *) setObjectWithInt:(NSInteger)value forKey:(NSString *)attrName;
- (void) setTemplateLib:(STGroup *)aTemplateLib;
- (STGroup *) getTemplateLib;
/* ----- ST end STAttrMap.methodsDecl ----- */
>>

@STAttrMap.methods() ::= <<
/* ----- ST start STAttrMap.methods ----- */
- (id) init
{
    self = [super initWithLen:16];
    if ( self != nil ) {
        templateLib = [STGroup newSTGroup:"<name>Templates"];
    }
    return self;
}

```

```

}

- (STAttrMap *) setObject:(id)aValue forKey:(NSString *)aAttrName
{
    [super setObject:aValue forKey:aAttrName];
    return self;
}

- (STAttrMap *) setObjectWithInt:(NSInteger)aValue forKey:(NSString *)aAttrName
{
    [super setObject:[NSNumber numberWithInt:aValue] forKey:aAttrName];
    return self;
}

- (void) setTemplateLib:(STGroup *)aTemplateLib
{
    templateLib = aTemplateLib;
}

- (STGroup *)getTemplateLib
{
    return templateLib;
}

/* ----- ST end STAttrMap.methods ----- */
>>

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ST ruleRefAndListLable */
<ruleRef(...)>
<listLabel(elem=[label getTemplate,...]>
>>

rewriteTemplate(alts) ::= <<
/* ----- ST start rewriteTemplate ----- */
// TEMPLATE REWRITE
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
    <alts:rewriteTemplateAlt(); separator="else ">
    <if(rewriteMode)><replaceTextInLine()><endif>
}
<else>
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
/* ----- ST end rewriteTemplate ----- */
>>

replaceTextInLine() ::= <<

```

```

/* ----- ST start replaceTextInLine ----- */
<if(TREE_PARSER)>
[[ (TokenRewriteStream *)input getTokenStream]
  replaceFromIndex: [[input getTreeAdaptor] getTokenStartIndex:retval.start]
    ToIndex: [[input getTreeAdaptor] getTokenStopIndex:retval.start]
    Text:retval.st];
<else>
[[ (TokenRewriteStream *)input)
  replaceFromIndex: [((CommonToken *)retval.start) getTokenIndex]
    ToIndex: [[input LT:-1] getTokenIndex]
    Text:retval.st];
<endif>
/* ----- ST end replaceTextInLine ----- */
>>

rewriteTemplateAlt() ::= <<
/* ----- ST start rewriteTemplateAlt ----- */
/* ST <it.description> */
<if(it.pred)>
if (<it.pred>) {
  retval.st = <it.alt>;
}<\n>
<else>
{
  retval.st = <it.alt>;
}<\n>
<endif>
/* ----- ST end rewriteTemplateAlt ----- */
>>

rewriteEmptyTemplate(alts) ::= <<
nil;
>>

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template *after* having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded
 * template.
 */
rewriteExternalTemplate(name,args) ::= <<
/* ----- ST start rewriteExternalTemplate ----- */
[templateLib getInstanceOf:@"<name>"]
<if(args)>[[STAttrMap newSTAttrMap] <args:{a | setObject:<a.value> forKey:@"<a.name>"}]><endif>]
/* ----- ST end rewriteExternalTemplate ----- */
>>

/** expr is a string expression that says what template to load */

```

```

rewriteIndirectTemplate(expr,args) ::= <<
/* ----- ST start rewriteIndirectTemplate ----- */
[templateLib getInstanceOf:<expr>
<if(args)> [[STAttrMap newSTAttrMap]<args:{a | setObject:<a.value> forKey:@"<a.name>"}]>]
<else>]<endif>
/* ----- ST end rewriteIndirectTemplate ----- */
>>

```

```

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <<
/* ----- ST start rewriteInlineTemplate ----- */
STGroup *templateLib;
templateLib.templates = [STAttrMap newSTAttrMap];
<if(args)> [templateLib.templates <args:{a | setObject:<a.value> forKey:@"<a.name>"};><endif>
[ST newST:templateLib template:@"<template>"];
/* ----- ST end rewriteInlineTemplate ----- */
>>

```

```

/** plain -> {foo} action */
rewriteAction(action) ::= <<
/* ----- ST start rewriteAction ----- */
<action>
/* ----- ST end rewriteAction ----- */
>>

```

```

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
/* ----- ST start actionSetAttribute ----- */
[[ST attributes] setObject:<expr> forKey:@"<attrName>"];
<![<st> setAttribute:<expr> name:@"<attrName>"];!>
/* ----- ST end actionSetAttribute ----- */
>>

```

```

/** Translate %{stringExpr} */
actionStringConstructor(stringExpr) ::= <<
/* ----- ST start actionStringConstructor ----- */
[ST newSTWithTemplate:<stringExpr>]
/* ----- ST end actionStringConstructor ----- */
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/ObjC/ST.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2010 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

```
javaTypeInitMap ::= [
  "int": "0",
  "long": "0",
  "float": "0.0f",
  "double": "0.0",
  "boolean": "false",
  "byte": "0",
  "short": "0",
  "char": "0",
  default: "null" // anything other than an atomic type
]
```

```
// System.Boolean.ToString() returns "True" and "False", but the proper C# literals are "true" and "false"
// The Java version of Boolean returns "true" and "false", so they map to themselves here.
```

```
booleanLiteral ::= [
  "True": "true",
  "False": "false",
  "true": "true",
  "false": "false",
  default: "false"
]
```

```
/** The overall file structure of a recognizer; stores methods for rules
```

```
* and cyclic DFAs plus support code.
```

```

*/
outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,
    docComment, recognizer,
    name, tokens, tokenNames, rules, cyclicDFAs,
    bitsets, buildTemplate, buildAST, rewriteMode, profile,
    backtracking, synpreds, memoize, numRules,
    fileName, ANTLRVersion, generatedTimestamp, trace,
    scopes, superClass, literals) ::=
<<
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
<actions.(actionScope).header>

<@imports>
import org antlr.runtime.*;
<if(TREE_PARSER)>
import org antlr.runtime.tree.*;
<endif>
import java.util.Stack;
import java.util.List;
import java.util.ArrayList;
<if(backtracking)>
import java.util.Map;
import java.util.HashMap;
<endif>
<@end>

<docComment>
@SuppressWarnings("all")
<recognizer>

>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode, labelType="CommonToken",
    superClass="Lexer") ::= <<
public class <grammar.recognizerName> extends <@superClassName><superClass><@end> {
    <tokens:{it | public static final int <it.name>=<it.type>;}; separator="\n">
    <scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>
    <actions.lexer.members>

    // delegates
    <grammar.delegates:
    {g|public <g.recognizerName> <g:delegateName()>;}; separator="\n">
    // delegators
    <grammar.delegators:
    {g|public <g.recognizerName> <g:delegateName()>;}; separator="\n">
    <last(grammar.delegators):{g|public <g.recognizerName> gParent;}>
    public <superClass>[] getDelegates() {
        return new <superClass>[] {<grammar.delegates: {g|<g:delegateName()>;}; separator = ", ">};

```

```

}

public <grammar.recognizerName>() {} <! needed by subclasses !>
public <grammar.recognizerName>(CharStream input<grammar.delegators:{g|, <g.recognizerName>
<g:delegateName()>>}) {
    this(input, new RecognizerSharedState()<grammar.delegators:{g|, <g:delegateName()>>});
}
public <grammar.recognizerName>(CharStream input, RecognizerSharedState state<grammar.delegators:{g|,
<g.recognizerName> <g:delegateName()>>}) {
    super(input,state);
<if(memoize)>
<if(grammar.grammarIsRoot)>
    state.ruleMemo = new HashMap[<numRules>+1];<\n><! index from 1..n !>
<endif>
<endif>
    <grammar.directDelegates:
        {g|<g:delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|, <p:delegateName()>}>,
this);}; separator="\n">
    <grammar.delegators:
        {g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
    <last(grammar.delegators):{g|gParent = <g:delegateName()>;}>
    }
    @Override public String getGrammarFileName() { return "<fileName>"; }

<if(filterMode)>
    <filteringNextToken()>
<endif>
    <rules; separator="\n\n">

    <synpreds:{p | <lexerSynpred(p)>}>

    <cyclicDFAs:{dfa | protected DFA<dfa.decisionNumber> dfa<dfa.decisionNumber> = new
DFA<dfa.decisionNumber>(this);}; separator="\n">
    <cyclicDFAs:cyclicDFA(); separator="\n\n"><! dump tables for all DFA !>

}
>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */
filteringNextToken() ::= <<
@Override
public Token nextToken() {
    while (true) {

```

```

if ( input.LA(1)==CharStream.EOF ) {
    Token eof = new CommonToken(input,Token.EOF,
        Token.DEFAULT_CHANNEL,
        input.index(),input.index());
    eof.setLine(getLine());
    eof.setCharPositionInLine(getCharPositionInLine());
    return eof;
}
state.token = null;
state.channel = Token.DEFAULT_CHANNEL;
state.tokenStartCharIndex = input.index();
state.tokenStartCharPositionInLine = input.getCharPositionInLine();
state.tokenStartLine = input.getLine();
state.text = null;
try {
    int m = input.mark();
    state.backtracking=1; <! means we won't throw slow exception !>
    state.failed=false;
    mTokens();
    state.backtracking=0;
    <! mTokens backtracks with synpred at backtracking==2
    and we set the synpredgate to allow actions at level 1. !>
    if ( state.failed ) {
        input.rewind(m);
        input.consume(); <! advance one char and try again !>
    }
    else {
        emit();
        return state.token;
    }
}
catch (RecognitionException re) {
    // shouldn't happen in backtracking mode, but...
    reportError(re);
    recover(re);
}
}

@Override
public void memoize(IntStream input,
    int ruleIndex,
    int ruleStartIndex)
{
    if ( state.backtracking>1 ) super.memoize(input, ruleIndex, ruleStartIndex);
}

@Override

```



```

public boolean alreadyParsedRule(IntStream input, int ruleIndex) {
if ( state.backtracking>1 ) return super.alreadyParsedRule(input, ruleIndex);
return false;
}
>>

actionGate() ::= "state.backtracking==0"

filteringActionGate() ::= "state.backtracking==1"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass,
    labelType, members, rewriteElementType,
    filterMode, ASTLabelType="Object") ::= <<
public class <grammar.recognizerName> extends <@superClassName><superClass><<@end> {
<if(grammar.grammarIsRoot)>
public static final String[] tokenNames = new String[] {
    "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ", wrap="\n\t">
};
<endif>
<tokens:{it |public static final int <it.name>=<it.type>;}; separator="\n">

// delegates
<grammar.delegates: {g|public <g.recognizerName> <g.delegateName()>;}; separator="\n">
public <superClass>[] getDelegates() {
return new <superClass>[] {<grammar.delegates: {g|<g.delegateName()>;}; separator = ", ">};
}

// delegators
<grammar.delegators:
{g|public <g.recognizerName> <g.delegateName()>;}; separator="\n">
<last(grammar.delegators):{g|public <g.recognizerName> gParent;}>

<scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>

<@members>
<! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
public <grammar.recognizerName><(<inputStreamType> input<grammar.delegators:{g|, <g.recognizerName>
<g.delegateName()>>}> {
this(input, new RecognizerSharedState()<grammar.delegators:{g|, <g.delegateName()>>});
}
public <grammar.recognizerName><(<inputStreamType> input, RecognizerSharedState
state<grammar.delegators:{g|, <g.recognizerName> <g.delegateName()>>}> {
super(input, state);
<parserCtorBody()>
<grammar.directDelegates:
{g|<g.delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|, <p.delegateName()>>},

```

```

this);}; separator="\n">
  <grammar.indirectDelegates:{ g | <g:delegateName()> = <g.delegateName()>.<g.delegateName()>;}
separator="\n">
  <last(grammar.delegators):{ g|gParent = <g:delegateName()>;}>
  }
<@end>

@Override public String[] getTokenNames() { return
<grammar.composite.rootGrammar.recognizerName>.tokenNames; }
@Override public String getGrammarFileName() { return "<fileName>"; }

<members>

<rules; separator="\n\n">

<! generate rule/method definitions for imported rules so they
  appear to be defined in this recognizer. !>
// Delegated rules
<grammar.delegatedRules:{ ruleDescriptor|
  public <returnType(ruleDescriptor)> <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope()>)
  throws <ruleDescriptor.throwsSpec; separator=", "> { <if(ruleDescriptor.hasReturnValue)>return
  <endif><ruleDescriptor.grammar:delegateName()>.<ruleDescriptor.name>(<if(ruleDescriptor.parameterScope)><ru
  leDescriptor.parameterScope.attributes:{ a|<a.name>; separator=", "><endif>); \}}; separator="\n">

  <synpreds:{ p | <synpred(p)>}>

  <cyclicDFAs:{ dfa | protected DFA<dfa.decisionNumber> dfa<dfa.decisionNumber> = new
  DFA<dfa.decisionNumber>(this);}; separator="\n">
  <cyclicDFAs:cyclicDFA(); separator="\n\n"><! dump tables for all DFA !>

  <bitsets:{ it | <bitset(name={ FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},
    words64=it.bits)>; separator="\n">
  }
  >>

  parserCtorBody() ::= <<
  <if(memoize)>
  <if(grammar.grammarIsRoot)>
  this.state.ruleMemo = new HashMap[<length(grammar.allImportedRules)>+1];<n><! index from 1..n !>
  <endif>
  <endif>
  <grammar.delegators:
  { g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
  >>

  parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,
    ASTLabelType="Object", superClass="Parser", labelType="Token",
    members={<actions.parser.members>}) ::= <<

```

```

<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, "TokenStream", superClass,
    labelType, members, "Token",
    false, ASTLabelType)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
    numRules, bitsets, filterMode, labelType={ <ASTLabelType> }, ASTLabelType="Object",
    superClass={ <if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif> },
    members={ <actions.treeparser.members> }
    ) ::= <<
<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, "TreeNodeStream", superClass,
    labelType, members, "Node",
    filterMode, ASTLabelType)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start <ruleName>
public final void <ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope(>) throws
<ruleDescriptor.throwsSpec: {x|<x>} ; separator=", "> {
    <ruleLabelDefs(>
<if(trace)>
    traceIn("<ruleName>_fragment", <ruleDescriptor.index>);
    try {
        <block>
    }
    finally {
        traceOut("<ruleName>_fragment", <ruleDescriptor.index>);
    }
<else>
    <block>
<endif>
}
// $ANTLR end <ruleName>
>>

```

```

synpred(name) ::= <<
public final boolean <name>() {
    state.backtracking++;
    <@start(>
int start = input.mark();
try {
    <name>_fragment(); // can never throw exception
} catch (RecognitionException re) {
    System.err.println("impossible: "+re);
}
boolean success = !state.failed;
input.rewind(start);
<@stop(>
state.backtracking--;
state.failed=false;
return success;
}<\n>
>>

lexerSynpred(name) ::= <<
<synpred(name)>
>>

ruleMemoization(name) ::= <<
<if(memoize)>
if ( state.backtracking>0 && alreadyParsedRule(input, <ruleDescriptor.index> ) ) {
    <returnStatement((( <ruleReturnValue(>)))> }
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>if (state.failed) <returnStatement((( <ruleReturnValue(>)))><endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (state.backtracking>0) { state.failed=true;
<returnStatement((( <ruleReturnValue(>)))>}<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(scope=ruleDescriptor.returnScope)>

```

```

// $ANTLR start "<ruleName>"
// <fileName>:<description>
<if(isPredefinedRewriteRule.(ruleName) && filterMode && buildAST)>
@Override
<endif>
public final <returnType(ruleDescriptor)> <ruleName>(<ruleDescriptor.parameterScope:parameterScope(>) throws
<ruleDescriptor	throwsSpec:{x|<x>}; separator=", "> {
  <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
  <ruleScopeSetUp(>
  <ruleDeclarations(>
  <ruleLabelDefs(>
  <ruleDescriptor.actions.init>
  <@preamble(>
  try {
    <ruleMemoization(name=ruleName)>
    <block>
    <ruleCleanUp(>
    <(ruleDescriptor.actions.after):execAction(>
  }
  <if(exceptions)>
    <exceptions:{e|<catch(decl=e.decl,action=e.action)><\n>}>
  <else>
  <if(!emptyRule)>
  <if(actions.(actionScope).rulecatch)>
    <actions.(actionScope).rulecatch>
  <else>
    catch (RecognitionException re) {
      reportError(re);
      recover(input,re);
      <@setErrorReturnValue(>
    }
  <endif>
  <endif>
  <endif>
  finally {
    // do for sure before leaving
    <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
    <memoize(>
    <ruleScopeCleanUp(>
    <finally>
  }
  <@postamble(>
  <returnStatement((({<ruleReturnValue(>}), false)>
  }
// $ANTLR end "<ruleName>"
>>

```

```
returnStatement(returnValue, force=true) ::= <%
```

```

<if(!isEmptyString.(returnValue))>
return <returnValue>;
<elseif(force)>
return;
<endif>
%>

catch(decl,action) ::= <<
catch (<e.decl>) {
<e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType(ruleDescriptor)> retval = new <returnType(ruleDescriptor)>();
retval.start = input.LT(1);
<elseif(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes: { a |
<a.type> <a.name> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
int <ruleDescriptor.name>_startIndex = input.index();
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: { it |<it>_stack.push(new <it>_scope());}; separator="\n">
<ruleDescriptor.ruleScope: { it |<it.name>_stack.push(new <it.name>_scope());}; separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: { it |<it>_stack.pop();}; separator="\n">
<ruleDescriptor.ruleScope: { it |<it.name>_stack.pop();}; separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: { it |<labelType> <it.label.text>=null;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: { it |List<Object> list_<it.label.text>=null;}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels: { ll|RuleReturnScope <ll.label.text> = null;}; separator="\n">

```

>>

```
lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
:{it |<labelType> <it.label.text>=null;}; separator="\n"
>
<ruleDescriptor.charLabels:{it |int <it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]
:{it |List<Object> list_<it.label.text>=null;}; separator="\n"
>
>>
```

```
ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
%>
```

```
ruleCleanUp() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.stop = input.LT(-1);
<endif>
<endif>
>>
```

```
memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( state.backtracking>0 ) { memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }
<endif>
<endif>
>>
```

```
/** How to generate a rule in the lexer; naked blocks are used for
* fragment rules.
*/
```

```
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
// $ANTLR start "<ruleName>"
```

```

public final void m<ruleName>(<ruleDescriptor.parameterScope:parameterScope()>) throws RecognitionException
{
    <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
    try {
    <if(nakedBlock)>
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block>
    <else>
        int _type = <ruleName>;
        int _channel = DEFAULT_TOKEN_CHANNEL;
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block>
        <ruleCleanUp()>
        state.type = _type;
        state.channel = _channel;
        <(ruleDescriptor.actions.after):execAction()>
    <endif>
    }
    finally {
        // do for sure before leaving
        <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
        <ruleScopeCleanUp()>
        <memoize()>
    }
}
// $ANTLR end "<ruleName>"
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
@Override
public void mTokens() throws RecognitionException {
    <block>
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

```



```

// <fileName>:<description>
int alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
<@prebranch()>
switch (alt<decisionNumber>) {
  <alts:{a | <altSwitchCase(i,a)>}>
}
<@postbranch()>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
int alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
switch (alt<decisionNumber>) {
  <alts:{a | <altSwitchCase(i,a)>}>
}
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
int cnt<decisionNumber>=0;

```

```

<decls>
<@preloop()>
loop<decisionNumber>:
while (true) {
int alt<decisionNumber>=<maxAlt>;
<@predecision()>
<decision>
<@postdecision()>
switch (alt<decisionNumber>) {
<alts:{ a | <altSwitchCase(i,a)>}>
default :
if ( cnt<decisionNumber> >= 1 ) break loop<decisionNumber>;
<ruleBacktrackFailure()>
EarlyExitException eee = new EarlyExitException(<decisionNumber>, input);
<@earlyExitException()>
throw eee;
}
cnt<decisionNumber>++;
}
<@postloop()>
>>

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */

closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=

<<

// <fileName>:<description>

<decls>

<@preloop()>

loop<decisionNumber>:

while (true) {

int alt<decisionNumber>=<maxAlt>;

<@predecision()>

<decision>

<@postdecision()>

switch (alt<decisionNumber>) {

<alts:{ a | <altSwitchCase(i,a)>}>

default :

break loop<decisionNumber>;

}

}

<@postloop()>

>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation

```

* so we can just use the normal block template
*/
optionalBlock ::= block

optionalBlockSingleAlt ::= block

/** A case in a switch that jumps to an alternative given the alternative
* number. A DFA predicts the alternative and then a simple switch
* does the jump to the code that actually matches that alternative.
*/
altSwitchCase(altNum,alt) ::= <<
case <altNum> :
  <@prealt()>
  <alt>
  break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
  <@declarations()>
  <elements:element()>
  <rew>
  <@cleanup()>
}
>>

/** What to emit when there is no rewrite. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(e) ::= <<
<@prematch()>
<e.el>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<if(label)><label>=<(<labelType>)<endif>match(input,<token>,FOLLOW_<token>_in_<ruleName><elementIndex
>); <checkRuleBacktrackFailure()>
>>

/** ids+=ID */

```

```

tokenRefAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<tokenRef(token,label,elementIndex,terminalOptions)>
<listLabel(label, label)>
>>

listLabel(label,elem) ::= <<
if (list_<label>==null) list_<label>=new ArrayList<Object>();
list_<label>.add(<elem>);
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = input.LA(1);
<endif>
match(<char>); <checkRuleBacktrackFailure()>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = input.LA(1);
<endif>
matchRange(<a>,<b>); <checkRuleBacktrackFailure()>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="",terminalOptions={ }) ::= <<
<if(label)>
<if(LEXER)>
<label>= input.LA(1);
<else>
<label>=<castToLabelType("input.LT(1)")>;
<endif>
<endif>
if ( <s> ) {
input.consume();
<postmatchCode>
<if(!LEXER)>
state.errorRecovery=false;
<endif>
<if(backtracking)>state.failed=false;<endif>
}
else {
<ruleBacktrackFailure()>
MismatchedSetException mse = new MismatchedSetException(null,input);
<@mismatchedSetException()>
<if(LEXER)>

```

```

recover(mse);
throw mse;
<else>
throw mse;
<! use following code to make it recover inline; remove throw mse;
recoverFromMismatchedSet(input,mse,FOLLOW_set_in_<ruleName><elementIndex>);
!>
<endif>
}
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(label, label)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex="0") ::= <<
<if(label)>
int <label>Start = getCharIndex();
match(<string>); <checkRuleBacktrackFailure()>
int <label>StartLine<elementIndex> = getLine();
int <label>StartCharPos<elementIndex> = getCharPositionInLine();
<label> = new <labelType>(input, Token.INVALID_TOKEN_TYPE, Token.DEFAULT_CHANNEL, <label>Start,
getCharIndex()-1);
<label>.setLine(<label>StartLine<elementIndex>);
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>);
<else>
match(<string>); <checkRuleBacktrackFailure()>
<endif>
>>

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<if(label)>
<label>=<castToLabelType("input.LT(1)")>;
<endif>
matchAny(input); <checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<wildcard(...)>
<listLabel(label, label)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<

```

```

<if(label)>
<label> = input.LA(1);
<endif>
matchAny(); <checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(label, elementIndex)>
<listLabel(label, label)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
pushFollow(FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)><scope.delegateName()>.<endif><rule.name>(<args; separator=", ">);
state._fsp--;
<checkRuleBacktrackFailure()>
>>

/** ids+=1 */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, label)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
int <label>Start<elementIndex> = getCharIndex();
int <label>StartLine<elementIndex> = getLine();
int <label>StartCharPos<elementIndex> = getCharPositionInLine();
<if(scope)><scope.delegateName()>.<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<label> = new <labelType>(input, Token.INVALID_TOKEN_TYPE, Token.DEFAULT_CHANNEL,
<label>Start<elementIndex>, getCharIndex()-1);
<label>.setLine(<label>StartLine<elementIndex>);
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>);
<else>

```

```

<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(rule,label,args,elementIndex,scope)>
<listLabel(label, label)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
int <label>Start<elementIndex> = getCharIndex();
int <label>StartLine<elementIndex> = getLine();
int <label>StartCharPos<elementIndex> = getCharPositionInLine();
match(EOF); <checkRuleBacktrackFailure()>
<labelType> <label> = new <labelType>(input, EOF, Token.DEFAULT_CHANNEL, <label>Start<elementIndex>,
getCharIndex()-1);
<label>.setLine(<label>StartLine<elementIndex>);
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>);
<else>
match(EOF); <checkRuleBacktrackFailure()>
<endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg()>"
recRuleArg()             ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==Token.DOWN ) {
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
match(input, Token.UP, null); <checkRuleBacktrackFailure()>
}
<else>
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>

```

```

match(input, Token.UP, null); <checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(pred,description)>) ) {
  <ruleBacktrackFailure()>
  throw new FailedPredicateException(input, "<ruleName>", "<description>");
}
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<stateNumber> = input.LA(<k>);
<edges; separator="\nelse ">
<if(!isTrue.(last(edges).labelExpr) && (!last(edges).predicates))>
else {
  <if(eotPredictsAlt)>
  alt<decisionNumber>=<eotPredictsAlt>;
  <else>
  <ruleBacktrackFailure()>
  <(nvaExceptionWrapperMap.(k))({NoViableAltException nvae =
  new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input);
  <@noViableAltException()>
  throw nvae;})>
  <endif>
  }
  <endif>
  >>

nvaExceptionWrapperMap ::= [
  "1":"wrapNvaExceptionForK1",
  "2":"wrapNvaExceptionForK2",
  default:"wrapNvaExceptionForKN"
]

wrapNvaExceptionForK1(exceptionCode) ::= <<
<exceptionCode>
>>

wrapNvaExceptionForK2(exceptionCode) ::= <<
int nvaeMark = input.mark();
try {
  input.consume();

```



```

<exceptionCode>
} finally {
input.rewind(nvaeMark);
}
>>

```

```

wrapNvaExceptionForKN(exceptionCode) ::= <<
int nvaeMark = input.mark();
try {
for (int nvaeConsume = 0; nvaeConsume \< <k> - 1; nvaeConsume++) {
input.consume();
}
<exceptionCode>
} finally {
input.rewind(nvaeMark);
}
>>

```

```

/** Same as a normal DFA state except that we don't examine lookahead
* for the bypass alternative. It delays error detection but this
* is faster, smaller, and more what people expect. For (X)? people
* expect "if ( LA(1)==X ) match(X);" and that's it.
*/

```

```

dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<stateNumber> = input.LA(<k>);
<edges; separator="\nelse ">
>>

```

```

/** A DFA state that is actually the loopback decision of a closure
* loop. If end-of-token (EOT) predicts any of the targets then it
* should act like a default clause (i.e., no error can be generated).
* This is used only in the lexer so that for ('a')* on the end of a rule
* anything other than 'a' predicts exiting.
*/

```

```

dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<stateNumber> = input.LA(<k>);
<edges; separator="\nelse ">
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber>=<eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else {
alt<decisionNumber>=<eotPredictsAlt>;
}
<endif>
<endif>
>>

```

```

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>";

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates><endif>) {
    <targetState>
}
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k>) ) {
<edges; separator="\n">
default:
<if(eotPredictsAlt)>
    alt<decisionNumber>=<eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>
    <(nvaExceptionWrapperMap.(k))({NoViableAltException nvae =
        new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input);
    <@noViableAltException()>
        throw nvae;})>
    <endif>
}
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k>) ) {
    <edges; separator="\n">
}
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k>) ) {
<edges; separator="\n">
<if(eotPredictsAlt)>
default:
    alt<decisionNumber>=<eotPredictsAlt>;
    break;
<endif>
}
>>

```

```

}
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels:{it |case <it>:}; separator="\n">
{
<targetState>
}
break;
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = dfa<decisionNumber>.predict(input);
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
static final String DFA<dfa.decisionNumber>_eotS =
"<dfa.javaCompressedEOT; wrap="\n\t">";
static final String DFA<dfa.decisionNumber>_eofS =
"<dfa.javaCompressedEOF; wrap="\n\t">";
static final String DFA<dfa.decisionNumber>_minS =
"<dfa.javaCompressedMin; wrap="\n\t">";
static final String DFA<dfa.decisionNumber>_maxS =
"<dfa.javaCompressedMax; wrap="\n\t">";
static final String DFA<dfa.decisionNumber>_acceptS =
"<dfa.javaCompressedAccept; wrap="\n\t">";
static final String DFA<dfa.decisionNumber>_specialS =
"<dfa.javaCompressedSpecial; wrap="\n\t">";
static final String[] DFA<dfa.decisionNumber>_transitionS = {
<dfa.javaCompressedTransition:{s|<s; wrap="\n\t">}; separator=",\n">
};

static final short[] DFA<dfa.decisionNumber>_eot =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_eotS);

```

```

static final short[] DFA<dfa.decisionNumber>_eof =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_eofS);
static final char[] DFA<dfa.decisionNumber>_min =
DFA.unpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_minS);
static final char[] DFA<dfa.decisionNumber>_max =
DFA.unpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_maxS);
static final short[] DFA<dfa.decisionNumber>_accept =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_acceptS);
static final short[] DFA<dfa.decisionNumber>_special =
DFA.unpackEncodedString(DFA<dfa.decisionNumber>_specialS);
static final short[][] DFA<dfa.decisionNumber>_transition;

static {
int numStates = DFA<dfa.decisionNumber>_transitionS.length;
DFA<dfa.decisionNumber>_transition = new short[numStates][];
for (int i=0; i<numStates; i++) {
DFA<dfa.decisionNumber>_transition[i] = DFA.unpackEncodedString(DFA<dfa.decisionNumber>_transitionS[i]);
}
}

protected class DFA<dfa.decisionNumber> extends DFA {

public DFA<dfa.decisionNumber>(BaseRecognizer recognizer) {
this.recognizer = recognizer;
this.decisionNumber = <dfa.decisionNumber>;
this.eot = DFA<dfa.decisionNumber>_eot;
this.eof = DFA<dfa.decisionNumber>_eof;
this.min = DFA<dfa.decisionNumber>_min;
this.max = DFA<dfa.decisionNumber>_max;
this.accept = DFA<dfa.decisionNumber>_accept;
this.special = DFA<dfa.decisionNumber>_special;
this.transition = DFA<dfa.decisionNumber>_transition;
}
@Override
public String getDescription() {
return "<dfa.description>";
}
<@errorMethod()>
<if(dfa.specialStateSTs)>
@Override
public int specialStateTransition(int s, IntStream _input) throws NoViableAltException {
<if(LEXER)>
IntStream input = _input;
<endif>
<if(PARSER)>
TokenStream input = (TokenStream)_input;
<endif>
<if(TREE_PARSER)>

```

```

TreeNodeStream input = (TreeNodeStream)_input;
<endif>
int _s = s;
switch ( s ) {
<dfa.specialStateSTs:{state |
case <i0> : <! compressed special state numbers 0..n-1 !>
<state>} ; separator="\n">
}
<if(backtracking)>
if (state.backtracking>0) {state.failed=true; return -1;}
<endif>
NoViableAltException nvae =
new NoViableAltException(getDescription(), <dfa.decisionNumber>, _s, input);
error(nvae);
throw nvae;
}
<endif>
}
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
int LA<decisionNumber>_<stateNumber> = input.LA(1);
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
int index<decisionNumber>_<stateNumber> = input.index();
input.rewind();
<endif>
s = -1;
<edges; separator="\nelse ">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>);
<endif>
if ( s>=0 ) return s;
break;
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif>) {s = <targetStateNumber>;}
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */

```

```

eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "<left>&&<right>"

orPredicates(operands) ::= "<operands; separator=\\|\\>"

notPredicate(pred) ::= "!(<evalPredicate(pred,{}>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber>==<atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "input.LA(<k>)==<atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
(LA<decisionNumber>_<stateNumber> >= <lower> && LA<decisionNumber>_<stateNumber> \<= <upper>)
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(input.LA(<k>) >= <lower>
&& input.LA(<k>) \<= <upper>)"

setTest(ranges) ::= <<
<ranges; separator="||">
>>

// A T T R I B U T E S

globalAttributeScope(scope) ::= <<
<if(scope.attributes)>
protected static class <scope.name>_scope {
<scope.attributes:{it |<it.decl>;}; separator="\n">
}
protected Stack\<<scope.name>_scope> <scope.name>_stack = new Stack\<<scope.name>_scope>();
<endif>
>>

ruleAttributeScope(scope) ::= <<
<if(scope)>

```

```

<if(scope.attributes)>
protected static class <scope.name>_scope {
  <scope.attributes:{it |<it.decl>;}; separator="\n">
}
protected Stack<<scope.name>_scope> <scope.name>_stack = new Stack<<scope.name>_scope>();
<endif>
<endif>
>>

```

```
returnStructName(r) ::= "<r.name>_return"
```

```

returnType(ruleDescriptor) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.grammar.recognizerName>.<ruleDescriptor:returnStructName()>
<elseif(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
%>

```

```

/** Generate the Java type associated with a single or multiple return
 * values.
 */

```

```

ruleLabelType(referencedRule) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
<returnScopeBaseType()>
<elseif(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
%>

```

```

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

```

```

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */

```

```

initValue(typeName) ::= <<
<javaTypeInitMap.(typeName)>
>>

```

```
/** Define a rule label including default value */
```

```

ruleLabelDef(label) ::= <%
<ruleLabelType(referencedRule=label.referencedRule)> <label.label.text> =

```

```

<initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;
%>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
public static class <ruleDescriptor.returnStructName()> extends <returnScopeBaseType()> {
<if(scope)><scope.attributes:{it |public <it.decl>;}; separator="\n"><endif>
<@ruleReturnMembers()>
};
<endif>
>>

returnScopeBaseType() ::= <%
<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope
%>

parameterScope(scope) ::= <<
<scope.attributes:{it |<it.decl>}; separator=", ">
>>

parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name> =<expr>";

scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack.elementAt(<scope>_stack.size()-<negIndex>-1).<attr.name>
<else>
<if(index)>
<scope>_stack.elementAt(<index>).<attr.name>
<else>
<scope>_stack.peek().<attr.name>
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack.elementAt(<scope>_stack.size()-<negIndex>-1).<attr.name> =<expr>;
<else>
<if(index)>
<scope>_stack.elementAt(<index>).<attr.name> =<expr>;
<else>
<scope>_stack.peek().<attr.name> =<expr>;
<endif>
<endif>
%>

```



```

<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like { $function.size()>0 && $function::name.equals("foo") }?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
(<scope>!=null?(<<returnType(referencedRule)>><scope>).<attr.name>:<initValue(attr.type)>)
<else>
<scope>
<endif>
%>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
%>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> =<expr>;
<else>
<attr.name> =<expr>;
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "<scope>!=null?<scope>.getText():null)"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>!=null?<scope>.getType():0)"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>!=null?<scope>.getLine():0)"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>!=null?<scope>.getCharPositionInLine():0)"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>!=null?<scope>.getChannel():0)"

```

```

tokenLabelPropertyRef_index(scope,attr) ::= "<scope>!=null?<scope>.getTokenIndex():0)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "<scope>!=null?Integer.valueOf(<scope>.getText()):0)"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>!=null?(<castToLabelType({<scope>.start})>):null)"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>!=null?(<castToLabelType({<scope>.stop})>):null)"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>!=null?(<ASTLabelType><scope>.getTree()):null)"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
<scope>!=null?(input.getTokenStream().toString(
input.getTreeAdaptor().getTokenStartIndex(<scope>.start),
input.getTreeAdaptor().getTokenStopIndex(<scope>.start))):null)
<else>
<scope>!=null?input.toString(<scope>.start,<scope>.stop):null)
<endif>
%>

ruleLabelPropertyRef_st(scope,attr) ::= "<scope>!=null?((StringTemplate)<scope>.getTemplate()):null)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::=
"<scope>!=null?<scope>.getType():0)"
lexerRuleLabelPropertyRef_line(scope,attr) ::=
"<scope>!=null?<scope>.getLine():0)"
lexerRuleLabelPropertyRef_pos(scope,attr) ::=
"<scope>!=null?<scope>.getCharPositionInLine():-1)"
lexerRuleLabelPropertyRef_channel(scope,attr) ::=
"<scope>!=null?<scope>.getChannel():0)"
lexerRuleLabelPropertyRef_index(scope,attr) ::=
"<scope>!=null?<scope>.getTokenIndex():0)"
lexerRuleLabelPropertyRef_text(scope,attr) ::=
"<scope>!=null?<scope>.getText():null)"
lexerRuleLabelPropertyRef_int(scope,attr) ::=
"<scope>!=null?Integer.valueOf(<scope>.getText()):0)"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "<castToLabelType(\"retval.start\")>"
rulePropertyRef_stop(scope,attr) ::= "<castToLabelType(\"retval.stop\")>"
rulePropertyRef_tree(scope,attr) ::= "retval.tree"
rulePropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
input.getTokenStream().toString(
input.getTreeAdaptor().getTokenStartIndex(retval.start),
input.getTreeAdaptor().getTokenStopIndex(retval.start))
<else>
input.toString(retval.start,input.LT(-1))

```

```

<endif>
%>
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "getText()"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(getCharIndex()-1)"
lexerRulePropertyRef_int(scope,attr) ::= "Integer.valueOf(<scope>.getText())"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>";
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>";

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <%
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
  <action>
}
<else>
<action>
<endif>
%>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
public static final BitSet <name> = new BitSet(new long[] { <words64: {it |<it>L}; separator="," > });
>>

codeFileExtension() ::= ".java"

true_value() ::= "true"
false_value() ::= "false"

isEmptyString ::= [
  "" : true,
  default : false
]

```

```
isTrue ::= [  
  "true" : true,  
  default : false  
]
```

```
isDefaultLabelType ::= [  
  "Token" : true,  
  default : false  
]
```

```
isPredefinedRewriteRule ::= [  
  "topdown" : true,  
  "bottomup" : true,  
  default : false  
]
```

```
castToLabelType(value) ::= <%  
<if(!isDefaultLabelType.(labelType))>  
(<labelType>  
<endif>  
<value>  
>%
```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Java/Java.stg

No license file was found, but licenses were detected in source scan.

```
/*  
 * [The "BSD license"]  
 * Copyright (c) 2011 Terence Parr  
 * All rights reserved.  
 *  
 * Conversion to C#:  
 * Copyright (c) 2011 Sam Harwell, Pixel Mine, Inc.  
 * All rights reserved.  
 *  
 * Redistribution and use in source and binary forms, with or without  
 * modification, are permitted provided that the following conditions  
 * are met:  
 * 1. Redistributions of source code must retain the above copyright  
 * notice, this list of conditions and the following disclaimer.  
 * 2. Redistributions in binary form must reproduce the above copyright  
 * notice, this list of conditions and the following disclaimer in the  
 * documentation and/or other materials provided with the distribution.  
 * 3. The name of the author may not be used to endorse or promote products  
 * derived from this software without specific prior written permission.  
 *
```

```

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

csharpVisibilityMap ::= [
  "private":"private",
  "protected":"protected",
  "public":"public",
  "fragment":"private",
  default:"private"
]

```

```

/** The overall file structure of a recognizer; stores methods for rules
* and cyclic DFAs plus support code.
*/

```

```

outputFile( LEXER,PARSER,TREE_PARSER, actionScope, actions,
  docComment, recognizer,
  name, tokens, tokenNames, rules, cyclicDFAs,
  bitsets, buildTemplate, buildAST, rewriteMode, profile,
  backtracking, synpreds, memoize, numRules,
  fileName, ANTLRVersion, generatedTimestamp, trace,
  scopes, superClass, literals) ::=

```

```
<<
```

```
//-----
```

```
// <auto-generated>
```

```
// This code was generated by a tool.
```

```
// ANTLR Version: <ANTLRVersion>
```

```
//
```

```
// Changes to this file may cause incorrect behavior and will be lost if
```

```
// the code is regenerated.
```

```
// </auto-generated>
```

```
//-----
```

```
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
```

```
// The variable 'variable' is assigned but its value is never used.
```

```
#pragma warning disable 219
```

```
// Unreachable code detected.
```

```
#pragma warning disable 162
```

```
// Missing XML comment for publicly visible type or member 'Type_or_Member'
```

```

#pragma warning disable 1591
// CLS compliance checking will not be performed on 'type' because it is not visible from outside this assembly.
#pragma warning disable 3019

<actions.(actionScope).header>

<@imports>
using System.Collections.Generic;
using Antlr.Runtime;
using Antlr.Runtime.Misc;
<if(TREE_PARSER)>
using Antlr.Runtime.Tree;
using RewriteRuleITokenStream = Antlr.Runtime.Tree.RewriteRuleTokenStream;
<endif>
<@end>
<if(actions.(actionScope).namespace)>
namespace <actions.(actionScope).namespace>
{
<endif>
<docComment>
<recognizer>
<if(actions.(actionScope).namespace)>

} // namespace <actions.(actionScope).namespace>
<endif>
>>

lexerInputStreamType() ::= <<
<actions.(actionScope).inputStreamType; null="ICharStream">
>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode, labelType="CommonToken",
superClass={ <if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Lexer
<endif>},
rewriteElementType={}, ASTLabelType={}) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<ANTLRVersion>")]
[System.CLSCompliant(false)]
<parserModifier(grammar=grammar, actions=actions)> partial class <grammar.recognizerName> :
<@superClassName><superClass><@end>
{
<tokens: {it|public const int <it.name; format="id">=<it.type>;}; separator="\n">
<scopes: {it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
<actions.lexer.members>

// delegates
<grammar.delegates:
{g|private <g.recognizerName> <g.delegateName()>;}; separator="\n">
// delegators

```

```

<grammar.delegates:
  {g|private <g.recognizerName> <g.delegateName(>);}; separator="\n">
<last(grammar.delegates):{g|private <g.recognizerName> gParent;}>

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<! needed by subclasses !>
{
  OnCreated();
}

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<lexerInputStreamType(>
input<grammar.delegates:{g|, <g.recognizerName> <g.delegateName(>}&> )
: this(input, new RecognizerSharedState(<grammar.delegates:{g|, <g.delegateName(>}&>))
{
}

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<lexerInputStreamType(> input,
RecognizerSharedState state<grammar.delegates:{g|, <g.recognizerName> <g.delegateName(>}&>))
: base(input, state)
{
<if(memoize)>
<if(grammar.grammarIsRoot)>
  state.ruleMemo = new System.Collections.Generic.Dictionary<int, int>[<numRules>+1];<\n><! index from 1..n !>
<endif>
<endif>
  <grammar.directDelegates:
    {g|<g.delegateName(> = new <g.recognizerName>(input, this.state<trunc(g.delegates):{p|,
<p:delegateName(>}&>, this);}; separator="\n">
  <grammar.delegates:
    {g|this.<g.delegateName(> = <g.delegateName(>);}; separator="\n">
  <last(grammar.delegates):{g|gParent = <g.delegateName(>};}>

  OnCreated();
}
public override string GrammarFileName { get { return "<fileName>"; } }

<if(grammar.hasDelegates)>
public override <lexerInputStreamType(> CharStream
{
  get
  {
    return base.CharStream;
  }
  set
  {
    base.CharStream = value;
    <grammar.directDelegates:
      {g|<g.delegateName(> = new <g.recognizerName>(input, state<trunc(g.delegates):{p|, <p:delegateName(>}&>,
this);}; separator="\n">

```

```

    <grammar.delegators:
    {g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
    <last(grammar.delegators):{g|gParent = <g:delegateName()>;}>
    }
    }

<if(grammar.delegates)>
public override void SetState(RecognizerSharedState state)
{
    base.SetState(state);
    <grammar.delegates:{g|<g:delegateName()>.SetState(state);}; separator="\n">
}
<endif>

<endif>
<if(filterMode)>
    <filteringNextToken()>
<endif>

partial void OnCreated();
partial void EnterRule(string ruleName, int ruleIndex);
partial void LeaveRule(string ruleName, int ruleIndex);

<rules; separator="\n">

<insertLexerSynpreds(synpreds)>

#region DFA
<cyclicDFAs:{dfa | DFA<dfa.decisionNumber> dfa<dfa.decisionNumber>;}; separator="\n">

protected override void InitDFAs()
{
    base.InitDFAs();
    <cyclicDFAs:{dfa | dfa<dfa.decisionNumber> = new DFA<dfa.decisionNumber>(this<if(dfa.specialStateSTs)>,
SpecialStateTransition<dfa.decisionNumber><endif>;}; separator="\n">
}

<cyclicDFAs:cyclicDFA()><! dump tables for all DFA !>
#endregion

}
>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens

```



```

* at backtracking==1.
*/
filteringNextToken() ::= <<
public override IToken NextToken()
{
while (true)
{
if (input.LA(1) == CharStreamConstants.EndOfFile)
{
IToken eof = new CommonToken((ICharStream)input, CharStreamConstants.EndOfFile, TokenChannels.Default,
input.Index, input.Index);
eof.Line = Line;
eof.CharPositionInLine = CharPositionInLine;
return eof;
}
state.token = null;
state.channel = TokenChannels.Default;
state.tokenStartCharIndex = input.Index;
state.tokenStartCharPositionInLine = input.CharPositionInLine;
state.tokenStartLine = input.Line;
state.text = null;
try
{
int m = input.Mark();
state.backtracking=1;<! means we won't throw slow exception !>
state.failed=false;
mTokens();
state.backtracking=0;
<! mTokens backtracks with synpred at backtracking==2
and we set the synpredgate to allow actions at level 1. !>
if (state.failed)
{
input.Rewind(m);
input.Consume();<! advance one char and try again !>
}
else
{
Emit();
return state.token;
}
}
catch (RecognitionException re)
{
// shouldn't happen in backtracking mode, but...
ReportError(re);
Recover(re);
}
}
}

```

```

}

public override void Memoize(IIntStream input, int ruleIndex, int ruleStartIndex)
{
    if (state.backtracking > 1)
        base.Memoize(input, ruleIndex, ruleStartIndex);
}

public override bool AlreadyParsedRule(IIntStream input, int ruleIndex)
{
    if (state.backtracking > 1)
        return base.AlreadyParsedRule(input, ruleIndex);

    return false;
}
>>

actionGate() ::= "state.backtracking == 0"

filteringActionGate() ::= "state.backtracking == 1"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass,
    labelType, members, rewriteElementType,
    filterMode, ASTLabelType="object") ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<ANTLRVersion>")]
[System.CLSCompliant(false)]
<parserModifier(grammar=grammar, actions=actions)> partial class <grammar.recognizerName> :
<@superClassName><superClass><@end>
{
<if(grammar.grammarIsRoot)>
internal static readonly string[] tokenNames = new string[] {
    "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ">
};
<endif>
<tokens:{it|public const int <it.name; format="id">=<it.type>;}; separator="\n">

<if(grammar.delegates)>
// delegates
<grammar.delegates:
    {g|private <g.recognizerName> <g.delegateName(>);}; separator="\n">
<endif>
<if(grammar.delegators)>
// delegators
<grammar.delegators:
    {g|private <g.recognizerName> <g.delegateName(>);}; separator="\n">
<last(grammar.delegators):{g|private <g.recognizerName> gParent;}>

```

```

<endif>

<if(grammar.delegates)>
public override void SetState(RecognizerSharedState state)
{
    base.SetState(state);
    <grammar.delegates: {g|<g:delegateName().SetState(state);}; separator="\n">
}

<if(TREE_PARSER)>
public override void SetTreeNodeStream(ITreeNodeStream input)
{
    base.SetTreeNodeStream(input);
    <grammar.delegates: {g|<g:delegateName().SetTreeNodeStream(input);}; separator="\n">
}
<endif>
<endif>

<scopes: {it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
<@members()>

public override string[] TokenNames { get { return
<grammar.composite.rootGrammar.recognizerName>.tokenNames; } }
public override string GrammarFileName { get { return "<fileName>"; } }

<members>

partial void OnCreated();
partial void EnterRule(string ruleName, int ruleIndex);
partial void LeaveRule(string ruleName, int ruleIndex);

#region Rules
<rules; separator="\n">
#endregion Rules

<if(grammar.delegatedRules)>
<! generate rule/method definitions for imported rules so they
appear to be defined in this recognizer. !>
#region Delegated rules
<grammar.delegatedRules: {ruleDescriptor|
    <ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> <return Type(ruleDescriptor)>
    <ruleDescriptor.name; format="id"><ruleDescriptor.parameterScope:parameterScope()><!throws
RecognitionException !>{ <if(ruleDescriptor.hasReturnValue)>return
<endif><ruleDescriptor.grammar:delegateName().<ruleDescriptor.name;
format="id"><ruleDescriptor.parameterScope><ruleDescriptor.parameterScope.attributes: {a|<a.name;
format="id">}; separator=", "><endif>}; \}}; separator="\n">
#endregion Delegated rules
<endif>

```

```

<insertSynpreds(synpreds)>

<if(cyclicDFAs)>
#region DFA
<cyclicDFAs:{ dfa | private DFA<dfa.decisionNumber> dfa<dfa.decisionNumber>;}; separator="\n">

protected override void InitDFAs()
{
    base.InitDFAs();
    <cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = new DFA<dfa.decisionNumber>( this<if(dfa.specialStateSTs)>,
SpecialStateTransition<dfa.decisionNumber><endif> );}; separator="\n">
}

<cyclicDFAs:cyclicDFA()><! dump tables for all DFA !>
#endregion DFA
<endif>

<if(bitsets)>
#region Follow sets
private static class Follow
{
    <bitsets:{ it|<bitset(name=_<it.name>_in_<it.inName><it.tokenIndex>), words64=it.bits}>; separator="\n">
}
#endregion Follow sets
<endif>
}
>>

@genericParser.members() ::= <<
<! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName><(inputStreamType>
input<grammar.delegates:{g|, <g.recognizerName> <g.delegateName()>>>)
: this(input, new RecognizerSharedState()<grammar.delegates:{g|, <g.delegateName()>>>)
{
}
}
<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName><(inputStreamType> input,
RecognizerSharedState state<grammar.delegates:{g|, <g.recognizerName> <g.delegateName()>>>)
: base(input, state)
{
<if(grammar.directDelegates)>
<grammar.directDelegates:
{g|<g.delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegates):{p|, <p.delegateName()>>},
this);}; separator="\n">
<endif>
<if(grammar.indirectDelegates)>
<grammar.indirectDelegates:{g | <g.delegateName()> = <g.delegator.delegateName()>.<g.delegateName()>};
separator="\n">

```

```

<endif>
<if(grammar.delegators)>
  <last(grammar.delegators):{g|gParent = <g:delegateName(>};}>
<endif>
<parserCtorBody()>
  OnCreated();
}
>>

// imported grammars are 'public' (can't be internal because their return scope classes must be accessible)
parserModifier(grammar, actions) ::= <<
<if(grammar.grammarIsRoot)><actions.(actionScope).modifier; null="public"><else>public<endif>
>>

parserCtorBody() ::= <<
<if(memoize)>
<if(grammar.grammarIsRoot)>
this.state.ruleMemo = new System.Collections.Generic.Dictionary<int,
int>[<length(grammar.allImportedRules)>+1];<n><! index from 1..n !>
<endif>
<endif>
<grammar.delegators:
{g|this.<g:delegateName(> = <g:delegateName(>}; separator="\n">
>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,
  ASTLabelType="object",
superClass={<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Parser
<endif>}, labelType="IToken",
  members={<actions.parser.members>}) ::= <<
<genericParser(inputStreamType="ITokenStream", rewriteElementType="IToken", filterMode=false, ...)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
  numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="object",
superClass={<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Tree.<
if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif><endif>},
  members={<actions.treeparser.members>}) ::= <<
<genericParser(inputStreamType="ITreeNodeStream", rewriteElementType="Node", ...)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.

```

```

* As predicates cannot be inlined into the invoking rule, they need to
* be in a rule by themselves.
*/
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<

partial void EnterRule_<ruleName>_fragment();
partial void LeaveRule_<ruleName>_fragment();

// $ANTLR start <ruleName>
<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> void
<ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope(>
{
  <ruleLabelDefs(...)>
  EnterRule_<ruleName>_fragment();
  EnterRule("<ruleName>_fragment", <ruleDescriptor.index>);
  TraceIn("<ruleName>_fragment", <ruleDescriptor.index>);
  try
  {
    <block>
  }
  finally
  {
    TraceOut("<ruleName>_fragment", <ruleDescriptor.index>);
    LeaveRule("<ruleName>_fragment", <ruleDescriptor.index>);
    LeaveRule_<ruleName>_fragment();
  }
}
// $ANTLR end <ruleName>
>>

insertLexerSynpreds(synpreds) ::= <<
<insertSynpreds(synpreds)>
>>

insertSynpreds(synpreds) ::= <<
<if(synpreds)>
#region Synpreds
private bool EvaluatePredicate(System.Action fragment)
{
  bool success = false;
  state.backtracking++;
  <@start(>
  try { DebugBeginBacktrack(state.backtracking);
  int start = input.Mark();
  try
  {
    fragment();

```

```

    }
    catch ( RecognitionException re )
    {
        System.Console.Error.WriteLine("impossible: "+re);
    }
    success = !state.failed;
    input.Rewind(start);
    } finally { DebugEndBacktrack(state.backtracking, success); }
    <@stop()>
    state.backtracking--;
    state.failed=false;
    return success;
    }
#endregion Synpreds
<endif>
>>

ruleMemoization(name) ::= <<
<if(memoize)>
if (state.backtracking > 0 && AlreadyParsedRule(input, <ruleDescriptor.index>)) { <returnFromRule()> }
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>if (state.failed) <returnFromRule()><endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (state.backtracking>0) { state.failed=true; <returnFromRule()>}<endif>
>>

ruleWrapperMap ::= [
    "bottomup":{<ruleWrapperBottomup()>},
    "topdown":{<ruleWrapperTopdown()>},
    default:""
]

ruleWrapperBottomup() ::= <<
<if(TREE_PARSER && filterMode)>
protected override <if(buildAST)>IAstRuleReturnScope<else>void<endif> Bottomup() { <if(buildAST)>return
<endif>bottomup(); }
<endif>
>>

ruleWrapperTopdown() ::= <<
<if(TREE_PARSER && filterMode)>

```

```

protected override <if(buildAST)>IAstRuleReturnScope<else>void<endif> Topdown() { <if(buildAST)>return
<endif>topdown(); }
<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(ruleDescriptor.returnScope)>
partial void EnterRule_<ruleName>();
partial void LeaveRule_<ruleName>();
<ruleWrapperMap.(ruleName)>
// $ANTLR start "<ruleName>"
// <fileName>:<description>
[GrammarRule("<ruleName>")]
<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> <returnType(ruleDescriptor)> <ruleName>;
format="id">(<ruleDescriptor.parameterScope:parameterScope(>))
{
  EnterRule_<ruleName>();
  EnterRule("<ruleName>", <ruleDescriptor.index>);
  TraceIn("<ruleName>", <ruleDescriptor.index>);
  <ruleScopeSetUp(>
  <ruleDeclarations(>
  <ruleLabelDefs(...)>
  <ruleDescriptor.actions.init>
  try { DebugEnterRule(GrammarFileName, "<ruleName>");
  DebugLocation(<ruleDescriptor.tree.line>, <ruleDescriptor.EORNode.charPositionInLine>);
  <@preamble(>
  try
  {
    <ruleMemoization(name=ruleName)>
    <block>
    <ruleCleanUp(>
    <(ruleDescriptor.actions.after):execAction(>
  }
  <if(exceptions)>
  <exceptions: {e|<catch(decl=e.decl,action=e.action)><\n>}>
  <else>
  <if(!emptyRule)>
  <if(actions.(actionScope).rulecatch)>
  <actions.(actionScope).rulecatch>
  <else>
  catch (RecognitionException re)
  {
    ReportError(re);
    Recover(input,re);

```



```

    <@setErrorReturnValue()>
    }
<endif>
<endif>
<endif>
finally
{
    TraceOut("<ruleName>", <ruleDescriptor.index>);
    LeaveRule("<ruleName>", <ruleDescriptor.index>);
    LeaveRule_<ruleName>();
    <memoize()>
    <ruleScopeCleanUp()>
    <finally>
    }
    DebugLocation(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.charPositionInLine>);
} finally { DebugExitRule(GrammarFileName, "<ruleName>"); }
<@postamble()>
<returnFromRule()><\n>
}
// $ANTLR end "<ruleName>"
>>

// imported grammars need to have internal rules
ruleModifier(grammar,ruleDescriptor) ::= <<
<if(grammar.grammarIsRoot)><csharpVisibilityMap.(ruleDescriptor.modifier);
null="private"><else>internal<endif>
>>

// imported grammars need to have public return scopes
returnScopeModifier(grammar,ruleDescriptor) ::= <<
<if(grammar.grammarIsRoot)><csharpVisibilityMap.(ruleDescriptor.modifier);
null="private"><else>public<endif>
>>

catch(decl,action) ::= <<
catch (<e.decl>)
{
    <e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType(ruleDescriptor)> retval = new
<returnType(ruleDescriptor)><(if(ruleDescriptor.returnScope.attributes)>this<endif>);
retval.Start = (<labelType>)input.LT(1);
<elseif(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes>: { a |

```

```

<a.type> <a.name; format="id"> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
int <ruleDescriptor.name>_startIndex = input.Index;
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: {it|<it>_stack.Push(new <it>_scope(this));<it>_scopeInit(<it>_stack.Peek());};
separator="\n">
<ruleDescriptor.ruleScope: {it|<it.name>_stack.Push(new
<it.name>_scope(this));<it.name>_scopeInit(<it.name>_stack.Peek());}; separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: {it|<it>_scopeAfter(<it>_stack.Peek());<it>_stack.Pop();}; separator="\n">
<ruleDescriptor.ruleScope: {it|<it.name>_scopeAfter(<it.name>_stack.Peek());<it.name>_stack.Pop();};
separator="\n">
>>

ruleLabelDefs(ruleDescriptor, labelType, ASTLabelType, rewriteElementType) ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it|<labelType> <it.label.text> = default(<labelType>);}; separator="\n"
>
<ruleDescriptor.tokenListLabels
: {it|List<<labelType>> list_<it.label.text> = null;}; separator="\n"
>
<[ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: {it|List<<ASTLabelType>> list_<it.label.text> = null;}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels:ruleLabelDef(); separator="\n">
>>

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it|<labelType> <it.label.text> = default(<labelType>);}; separator="\n"
>
<[ruleDescriptor.charListLabels,
ruleDescriptor.charLabels]
: {it|int <it.label.text> = 0;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]

```

```

: {it|List|<<labelType>> list_<it.label.text> = null;}; separator="\n"
>
<ruleDescriptor.charListLabels: {it|List|<int> list_<it.label.text> = null;}; separator="\n"
>
>>

returnFromRule() ::= <%
return
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<! This comment is a hack to make sure the following
  single space appears in the output. !> <ruleDescriptor.singleValueReturnName>
<else>
<!!> retval
<endif>
<endif>
<endif>
;
%>

ruleCleanUp() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.Stop = (<labelType>)input.LT(-1);
<endif>
<endif>
>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if (state.backtracking > 0) { Memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_startIndex); }
<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<

partial void EnterRule_<ruleName>();
partial void LeaveRule_<ruleName>();

// $ANTLR start "<ruleName>"
[GrammarRule("<ruleName>")]
<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> void

```

```

m<ruleName>(<ruleDescriptor.parameterScope:parameterScope()>)
{
  EnterRule_<ruleName>();
  EnterRule("<ruleName>", <ruleDescriptor.index>);
  TraceIn("<ruleName>", <ruleDescriptor.index>);
  <ruleScopeSetUp()>
  <ruleDeclarations()>
  try
  {
    <if(nakedBlock)>
      <ruleMemoization(name=ruleName)>
      <lexerRuleLabelDefs()>
      <ruleDescriptor.actions.init>
      <block>
    <else>
      int _type = <ruleName>;
      int _channel = DefaultTokenChannel;
      <ruleMemoization(name=ruleName)>
      <lexerRuleLabelDefs()>
      <ruleDescriptor.actions.init>
      <block>
      <ruleCleanUp()>
      state.type = _type;
      state.channel = _channel;
      <(ruleDescriptor.actions.after):execAction()>
    <endif>
  }
  finally
  {
    TraceOut("<ruleName>", <ruleDescriptor.index>);
    LeaveRule("<ruleName>", <ruleDescriptor.index>);
    LeaveRule_<ruleName>();
    <ruleScopeCleanUp()>
    <memoize()>
  }
}
// $ANTLR end "<ruleName>"
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<

public override void mTokens()
{
  <block><\n>
}

```

>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */

block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

int alt<decisionNumber>=<maxAlt>;

<decls>

<@predecision()>

try { DebugEnterSubRule(<decisionNumber>);

try { DebugEnterDecision(<decisionNumber>, false<!<decision.dfa.hasSynPred>!>);

<decision>

} finally { DebugExitDecision(<decisionNumber>); }

<@postdecision()>

<@prebranch()>

switch (alt<decisionNumber>)

{

<alts:{a|altSwitchCase(i,a)}>>

}

} finally { DebugExitSubRule(<decisionNumber>); }

<@postbranch()>

>>

/** A rule block with multiple alternatives */

ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

int alt<decisionNumber>=<maxAlt>;

<decls>

<@predecision()>

try { DebugEnterDecision(<decisionNumber>, false<!<decision.dfa.hasSynPred>!>);

<decision>

} finally { DebugExitDecision(<decisionNumber>); }

<@postdecision()>

switch (alt<decisionNumber>)

{

<alts:{a|altSwitchCase(i,a)}>>

}

>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

// <fileName>:<description>

<decls>

<@prealt()>

DebugEnterAlt(1);

<alts>

<@postalt()>

>>

```

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
DebugEnterAlt(1);
<alts>
<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
int cnt<decisionNumber>=0;
<decls>
<@preloop()>
try { DebugEnterSubRule(<decisionNumber>);
while (true)
{
int alt<decisionNumber>=<maxAlt>;
<@predecision()>
try { DebugEnterDecision(<decisionNumber>, false!<decision.dfa.hasSynPred!>);
<decision>
} finally { DebugExitDecision(<decisionNumber>); }
<@postdecision()>
switch (alt<decisionNumber>)
{
<alts:{a|<altSwitchCase(i,a)>}>
default:
if (cnt<decisionNumber> >= 1)
goto loop<decisionNumber>;

<ruleBacktrackFailure()>
EarlyExitException eee<decisionNumber> = new EarlyExitException( <decisionNumber>, input );
DebugRecognitionException(eee<decisionNumber>);
<@earlyExitException()>
throw eee<decisionNumber>;
}
cnt<decisionNumber>++;
}
loop<decisionNumber>:
;

} finally { DebugExitSubRule(<decisionNumber>); }
<@postloop()>
>>

```

```
positiveClosureBlockSingleAlt ::= positiveClosureBlock
```

```
/** A (..)* block with 1 or more alternatives */
```

```
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
```

```
<<
```

```
// <fileName>:<description>
```

```
<decls>
```

```
<@preloop()>
```

```
try { DebugEnterSubRule(<decisionNumber>);
```

```
while (true)
```

```
{
```

```
int alt<decisionNumber>=<maxAlt>;
```

```
<@predecision()>
```

```
try { DebugEnterDecision(<decisionNumber>, false<!<decision.dfa.hasSynPred!>);
```

```
<decision>
```

```
} finally { DebugExitDecision(<decisionNumber>); }
```

```
<@postdecision()>
```

```
switch ( alt<decisionNumber> )
```

```
{
```

```
<alts:{a|<altSwitchCase(i,a)>}>
```

```
default:
```

```
goto loop<decisionNumber>;
```

```
}
```

```
}
```

```
loop<decisionNumber>:
```

```
;
```

```
} finally { DebugExitSubRule(<decisionNumber>); }
```

```
<@postloop()>
```

```
>>
```

```
closureBlockSingleAlt ::= closureBlock
```

```
/** Optional blocks (x)? are translated to (x|) by before code generation
```

```
* so we can just use the normal block template
```

```
*/
```

```
optionalBlock ::= block
```

```
optionalBlockSingleAlt ::= block
```

```
/** A case in a switch that jumps to an alternative given the alternative
```

```
* number. A DFA predicts the alternative and then a simple switch
```

```
* does the jump to the code that actually matches that alternative.
```

```
*/
```

```
altSwitchCase(altNum,alt) ::= <<
```

```
case <altNum>:
```

```

<@prealt()>
DebugEnterAlt(<altNum>);
<alt>
break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
}
>>

/** What to emit when there is no rewrite. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(it) ::= <%
<@prematch()>
DebugLocation(<it.line>, <it.pos>);<\n>
<it.el><\n>
%>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<if(label)><label>=<labelType><endif>Match(input,<token>,Follow._<token>_in_<ruleName><elementIndex>);
<checkRuleBacktrackFailure()>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

listLabel(label,elem) ::= <<
#error The listLabel template should not be used with this target.<\n>
>>

listLabelElem(label,elem,elemType) ::= <<

```



```

if (list_<label>==null) list_<label>=new List<<elemType; null={ <labelType>}>>();
list_<label>.Add(<elem>);<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
Match(<char>); <checkRuleBacktrackFailure()>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
MatchRange(<a>,<b>); <checkRuleBacktrackFailure()>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="",terminalOptions={}) ::= <<
<if(label)>
<matchSetLabel()>
<endif>
if (<s>)
{
input.Consume();
<postmatchCode>
<if(!LEXER)>state.errorRecovery=false;<endif><if(backtracking)>state.failed=false;<endif>
}
else
{
<ruleBacktrackFailure()>
MismatchedSetException mse = new MismatchedSetException(null,input);
DebugRecognitionException(mse);
<@mismatchedSetException()>
<if(LEXER)>
Recover(mse);
throw mse;
<else>
throw mse;
<! use following code to make it recover inline; remove throw mse;
recoverFromMismatchedSet(input,mse,Follow._set_in_<ruleName><elementIndex>);
!>
<endif>
}<\n>
>>

```

```

matchSetUnchecked(s,label,elementIndex,postmatchCode=false) ::= <%
<if(label)>
<matchSetLabel()><\n>
<endif>
input.Consume();<\n>
<if(postmatchCode)>
<postmatchCode><\n>
<endif>
<if(!LEXER)>state.errorRecovery=false;<endif><if(backtracking)>state.failed=false;<endif>
%>

matchSetLabel() ::= <%
<if(LEXER)>
<label>= input.LA(1);
<else>
<label>=(<labelType>)input.LT(1);
<endif>
%>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <%
<if(label)>
int <label>Start = CharIndex;<\n>
Match(<string>); <checkRuleBacktrackFailure()><\n>
int <label>StartLine<elementIndex> = Line;<\n>
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
<label> = new <labelType>(input, TokenTypes.Invalid, TokenChannels.Default, <label>Start, CharIndex-1);<\n>
<label>.Line = <label>StartLine<elementIndex>;<\n>
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
<else>
Match(<string>); <checkRuleBacktrackFailure()><\n>
<endif>
%>

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<if(label)>
<label>=(<labelType>)input.LT(1);<\n>
<endif>
MatchAny(input); <checkRuleBacktrackFailure()>
>>

```

```

wildcardAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<wildcard(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
MatchAny(); <checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
PushFollow(Follow._<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)><scope.delegateName()>.<endif><rule.name; format="id">(<args;
separator=", ">);
PopFollow();
<checkRuleBacktrackFailure()>
>>

/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabelElem(elem=label,elemType={<ASTLabelType>},...)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <%
<if(label)>
int <label>Start<elementIndex> = CharIndex;<\n>

```

```

int <label>StartLine<elementIndex> = Line;<\n>
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()><\n>
<label> = new <labelType>(input, TokenType.Invalid, TokenChannels.Default, <label>Start<elementIndex>,
CharIndex-1);<\n>
<label>.Line = <label>StartLine<elementIndex>;<\n>
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
<else>
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<endif>
%>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <%
<if(label)>
int <label>Start<elementIndex> = CharIndex;<\n>
int <label>StartLine<elementIndex> = Line;<\n>
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
Match(EOF); <checkRuleBacktrackFailure()><\n>
<labelType> <label> = new <labelType>(input, EOF, TokenChannels.Default, <label>Start<elementIndex>,
CharIndex-1);<\n>
<label>.Line = <label>StartLine<elementIndex>;<\n>
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
<else>
Match(EOF); <checkRuleBacktrackFailure()>
<endif>
%>

// used for left-recursive rules
recRuleDefArg()           ::= "int <recRuleArg()>"
recRuleArg()              ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg() \<= <opPrec>"
recRuleSetResultAction()  ::= "root_0=$<ruleName>_primary.tree;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>

```

```

if (input.LA(1) == TokenTypes.Down)
{
    Match(input, TokenTypes.Down, null); <checkRuleBacktrackFailure()>
    <children:element()>
    Match(input, TokenTypes.Up, null); <checkRuleBacktrackFailure()>
}
<else>
Match(input, TokenTypes.Down, null); <checkRuleBacktrackFailure()>
<children:element()>
Match(input, TokenTypes.Up, null); <checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if (!(<evalPredicate(...)>))
{
    <ruleBacktrackFailure()>
    throw new FailedPredicateException(input, "<ruleName>", "<description>");
}
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<n>
<edges; separator="\nelse ">
else
{
    <if(eotPredictsAlt)>
    alt<decisionNumber> = <eotPredictsAlt>;
    <else>
    <ruleBacktrackFailure()>
    NoViableAltException nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>,
input, <k>);
    DebugRecognitionException(nvae);
    <@noViableAltException()>
    throw nvae;
    <endif>
}
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.

```

```

*/
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<\n>
<edges; separator="\nelse ">
>>

/** A DFA state that is actually the loopback decision of a closure
* loop. If end-of-token (EOT) predicts any of the targets then it
* should act like a default clause (i.e., no error can be generated).
* This is used only in the lexer so that for ('a'* on the end of a rule
* anything other than 'a' predicts exiting.
*/
*/
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<\n>
<edges; separator="\nelse "><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber> = <eotPredictsAlt>;<! if no edges, don't gen ELSE !>
<else>
else
{
alt<decisionNumber> = <eotPredictsAlt>;
}<\n>
<endif>
<endif>
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber> = <alt>;"

/** A simple edge with an expression. If the expression is satisfied,
* enter to the target state. To handle gated productions, we may
* have to evaluate some predicates for this edge.
*/
*/
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ((<labelExpr>)<if(predicates)> && (<predicates>)<endif>)
{
<targetState>
}
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
* decides if this is possible: CodeGenerator.canGenerateSwitch().
*/
*/
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch (input.LA(<k>))

```

```

{
<edges; separator="\n">
default:
<if(eotPredictsAlt)>
alt<decisionNumber>=<eotPredictsAlt>;
break;<\n>
<else>
{
<ruleBacktrackFailure()>
NoViableAltException nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>,
input, <k>);
DebugRecognitionException(nvae);
<@noViableAltException()>
throw nvae;
}
<endif>
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch (input.LA(<k>))
{
<edges; separator="\n">
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch (input.LA(<k>))
{
<edges; separator="\n">
<if(eotPredictsAlt)>
default:
alt<decisionNumber>=<eotPredictsAlt>;
break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels: {it|case <it>:}; separator="\n">
{
<targetState>
}
break;
>>

// C y c l i c D F A

```

```

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
try
{
alt<decisionNumber> = dfa<decisionNumber>.Predict(input);
}
catch (NoViableAltException nvae)
{
DebugRecognitionException(nvae);
throw;
}
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
private class DFA<dfa.decisionNumber> : DFA
{
private const string DFA<dfa.decisionNumber>_eotS =
"<dfa.javaCompressedEOT; wrap="+\n\t\t\t"">";
private const string DFA<dfa.decisionNumber>_eofS =
"<dfa.javaCompressedEOF; wrap="+\n\t\t\t"">";
private const string DFA<dfa.decisionNumber>_minS =
"<dfa.javaCompressedMin; wrap="+\n\t\t\t"">";
private const string DFA<dfa.decisionNumber>_maxS =
"<dfa.javaCompressedMax; wrap="+\n\t\t\t"">";
private const string DFA<dfa.decisionNumber>_acceptS =
"<dfa.javaCompressedAccept; wrap="+\n\t\t\t"">";
private const string DFA<dfa.decisionNumber>_specialS =
"<dfa.javaCompressedSpecial; wrap="+\n\t\t\t"">}>";
private static readonly string[] DFA<dfa.decisionNumber>_transitionS =
{
<dfa.javaCompressedTransition:{s|"<s; wrap="+\n\t\t\t"">"; separator=",\n">
};

private static readonly short[] DFA<dfa.decisionNumber>_eot =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_eotS);
private static readonly short[] DFA<dfa.decisionNumber>_eof =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_eofS);

```



```

private static readonly char[] DFA<dfa.decisionNumber>_min =
DFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_minS);
private static readonly char[] DFA<dfa.decisionNumber>_max =
DFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_maxS);
private static readonly short[] DFA<dfa.decisionNumber>_accept =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_acceptS);
private static readonly short[] DFA<dfa.decisionNumber>_special =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_specialS);
private static readonly short[][] DFA<dfa.decisionNumber>_transition;

static DFA<dfa.decisionNumber>()
{
    int numStates = DFA<dfa.decisionNumber>_transitionS.Length;
    DFA<dfa.decisionNumber>_transition = new short[numStates][];
    for ( int i=0; i < numStates; i++ )
    {
        DFA<dfa.decisionNumber>_transition[i] =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_transitionS[i]);
    }
}

public DFA<dfa.decisionNumber>( BaseRecognizer recognizer<if(dfa.specialStateSTs)>,
SpecialStateTransitionHandler specialStateTransition<endif> )
<if(dfa.specialStateSTs)>
    : base(specialStateTransition)
<endif>
{
    this.recognizer = recognizer;
    this.decisionNumber = <dfa.decisionNumber>;
    this.eot = DFA<dfa.decisionNumber>_eot;
    this.eof = DFA<dfa.decisionNumber>_eof;
    this.min = DFA<dfa.decisionNumber>_min;
    this.max = DFA<dfa.decisionNumber>_max;
    this.accept = DFA<dfa.decisionNumber>_accept;
    this.special = DFA<dfa.decisionNumber>_special;
    this.transition = DFA<dfa.decisionNumber>_transition;
}

public override string Description { get { return "<dfa.description>"; } }

public override void Error(NoViableAltException nvae)
{
    DebugRecognitionException(nvae);
}
}<\n>
<if(dfa.specialStateSTs)>
private int SpecialStateTransition<dfa.decisionNumber>(DFA dfa, int s, IIntStream _input)<! throws
NoViableAltException!>

```

```

{
<if(LEXER)>
  IIntStream input = _input;
<endif>
<if(PARSER)>
  ITokenStream input = (ITokenStream)_input;
<endif>
<if(TREE_PARSER)>
  ITreeNodeStream input = (ITreeNodeStream)_input;
<endif>
  int _s = s;
  s = -1;
  <! pull these outside the switch cases to save space on locals !>
  int LA<dfa.decisionNumber>_1 = input.LA(1);
  int index<dfa.decisionNumber>_1 = input.Index;
  switch (_s)
  {
  <dfa.specialStateSTs: {state |case <i0>:<! compressed special state numbers 0..n-1 !>
  <state>}; separator="\n">

  default:
    break;
  }

  if (s >= 0)
    return s;

  <if(backtracking)>
  if (state.backtracking > 0) {state.failed=true; return -1;}
  <endif>
  NoViableAltException nvae = new NoViableAltException(dfa.Description, <dfa.decisionNumber>, _s, input);
  dfa.Error(nvae);
  throw nvae;
  }
  <endif>
  >>

  /** A state in a cyclic DFA; it's a special state and part of a big switch on
  * state.
  */
  cyclicDFASState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
  {
  <if(semPredState)>
  <! get next lookahead symbol to test edges, then rewind !>
  input.Rewind();
  <endif>
  <edges; separator="\nelse ">
  <if(semPredState)>

```

```

<! return input cursor to state before we rewound !>
input.Seek(index<decisionNumber>_1);
<endif>
break;
}
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ((<labelExpr><if(predicates)> && (<predicates><endif>) {s = <targetStateNumber>;}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "(<left>&&<right>)"

orPredicates(operands) ::= "(<operands; separator=\\|\\>)"

notPredicate(pred) ::= "!(<evalPredicate(...)>)"

evalPredicate(pred,description) ::= "(<pred>)"

evalSynPredicate(pred,description) ::= "EvaluatePredicate(<pred>_fragment)"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<k>===<atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "input.LA(<k>)==<atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
(LA<decisionNumber>_<k><ge()><lower> && LA<decisionNumber>_<k><le()><upper>)
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::=
"(input.LA(<k><ge()><lower> && input.LA(<k><le()><upper>))"

```

```
le() ::= "\<="
```

```
ge() ::= ">="
```

```
setTest(ranges) ::= <<  
<ranges; separator="||">  
>>
```

```
// A T T R I B U T E S
```

```
attributeScope(scope) ::= <<
```

```
<if(scope)>
```

```
<if(scope.attributes)>
```

```
protected sealed partial class <scope.name>_scope
```

```
{
```

```
<scope.attributes:{it|public <it.decl>;}; separator="\n">
```

```
public <scope.name>_scope(<grammar.recognizerName> grammar) { OnCreated(grammar); }
```

```
partial void OnCreated(<grammar.recognizerName> grammar);
```

```
}
```

```
<if(scope.actions.scopeinit)>
```

```
protected void <scope.name>_scopeInit( <scope.name>_scope scope )
```

```
{
```

```
<scope.actions.scopeinit>
```

```
}
```

```
<else>
```

```
partial void <scope.name>_scopeInit( <scope.name>_scope scope );
```

```
<endif>
```

```
<if(scope.actions.scopeafter)>
```

```
protected void <scope.name>_scopeAfter( <scope.name>_scope scope )
```

```
{
```

```
<scope.actions.scopeafter>
```

```
}
```

```
<else>
```

```
partial void <scope.name>_scopeAfter( <scope.name>_scope scope );
```

```
<endif>
```

```
protected readonly ListStack<<scope.name>_scope> <scope.name>_stack = new
```

```
ListStack<<scope.name>_scope>();
```

```
<endif>
```

```
<endif>
```

```
>>
```

```
globalAttributeScope(scope) ::= <<
```

```
<attributeScope(...)>
```

```
>>
```

```
ruleAttributeScope(scope) ::= <<
```

```
<attributeScope(...)>
```

```
>>
```

```

returnStructName(it) ::= "<it.name>_return"

returnType(ruleDescriptor) ::= <%
<if(ruleDescriptor.returnScope.attributes && ruleDescriptor.hasMultipleReturnValues)>
  <ruleDescriptor.grammar.recognizerName>.<ruleDescriptor:returnStructName()>
<elseif(ruleDescriptor.hasMultipleReturnValues)>
  <ruleReturnBaseType()>
<elseif(ruleDescriptor.hasSingleReturnValue)>
  <ruleDescriptor.singleValueReturnType>
<else>
  void
<endif>
%>

/** Generate the C# type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
  <ruleReturnBaseType()>
<elseif(referencedRule.hasSingleReturnValue)>
  <referencedRule.singleValueReturnType>
<else>
  void
<endif>
%>

delegateName(it) ::= <<
<if(it.label)><it.label><else>g<it.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
default(<typeName>)
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <%
<ruleLabelType(label.referencedRule)> <label.label.text> = <initValue(ruleLabelType(label.referencedRule))>;
%>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */

```

```

returnScope(scope) ::= <<
<if(scope.attributes && ruleDescriptor.hasMultipleReturnValues)>
<returnScopeModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> sealed partial class
<ruleDescriptor:returnStructName()> : <ruleReturnBaseType()><@ruleReturnInterfaces()>
{
<scope.attributes:{it|public <it.decl>;}; separator="\n">
<@ruleReturnMembers()>
}
<endif>
>>

ruleReturnBaseType() ::= <%
<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope\<<labelType>>
%>

@returnScope.ruleReturnMembers() ::= <<
public <ruleDescriptor:returnStructName()>( <grammar.recognizerName> grammar) { OnCreated(grammar);}
partial void OnCreated( <grammar.recognizerName> grammar);
>>

parameterScope(scope) ::= <<
<scope.attributes:{it|<it.decl>}; separator=", ">
>>

parameterAttributeRef(attr) ::= <<
<attr.name; format="id">
>>

parameterSetAttributeRef(attr,expr) ::= <<
<attr.name; format="id"> =<expr>;
>>

scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack[<scope>_stack.Count - <negIndex> - 1].<attr.name; format="id">
<else>
<if(index)>
<scope>_stack[<index>].<attr.name; format="id">
<else>
<scope>_stack.Peek().<attr.name; format="id">
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack[<scope>_stack.Count - <negIndex> - 1].<attr.name; format="id"> = <expr>;
<else>

```

```

<if(index)>
<scope>_stack[<index>].<attr.name; format="id"> = <expr>;
<else>
<scope>_stack.Peek().<attr.name; format="id"> = <expr>;
<endif>
<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.Count>0 && $function::name.Equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
(<scope>!=null?(<<return Type(referencedRule)>><scope>).<attr.name; format="id">:<initValue(attr.type)>)
<else>
<scope>
<endif>
%>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name; format="id">
<else>
<attr.name; format="id">
<endif>
%>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name; format="id"> =<expr>;
<else>
<attr.name; format="id"> =<expr>;
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

```

```

tokenLabelPropertyRef_text(scope,attr) ::= "<scope>!=null?<scope>.Text:default(string)"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>!=null?<scope>.Type:0"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>!=null?<scope>.Line:0"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>!=null?<scope>.CharPositionInLine:0"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>!=null?<scope>.Channel:0"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>!=null?<scope>.TokenIndex:0"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "<scope>!=null?int.Parse(<scope>.Text):0"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>!=null?((<labelType><scope>.Start):default(<labelType>))"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>!=null?((<labelType><scope>.Stop):default(<labelType>))"
ruleLabelPropertyRef_tree(scope,attr) ::=
"<scope>!=null?((<ASTLabelType><scope>.Tree):default(<ASTLabelType>))"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
(<scope>!=null?(input.TokenStream.ToString(
input.TreeAdaptor.GetTokenStartIndex(<scope>.Start),
input.TreeAdaptor.GetTokenStopIndex(<scope>.Start))):default(string)
<else>
(<scope>!=null?input.ToString(<scope>.Start,<scope>.Stop):default(string))
<endif>
%>

ruleLabelPropertyRef_st(scope,attr) ::= "<scope>!=null?<scope>.Template:null"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::=
"<scope>!=null?<scope>.Type:0"

lexerRuleLabelPropertyRef_line(scope,attr) ::=
"<scope>!=null?<scope>.Line:0"

lexerRuleLabelPropertyRef_pos(scope,attr) ::=
"<scope>!=null?<scope>.CharPositionInLine:-1"

lexerRuleLabelPropertyRef_channel(scope,attr) ::=
"<scope>!=null?<scope>.Channel:0"

lexerRuleLabelPropertyRef_index(scope,attr) ::=
"<scope>!=null?<scope>.TokenIndex:0"

lexerRuleLabelPropertyRef_text(scope,attr) ::=
"<scope>!=null?<scope>.Text:default(string)"

lexerRuleLabelPropertyRef_int(scope,attr) ::=
"<scope>!=null?int.Parse(<scope>.Text):0"

```



```

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.Start"
rulePropertyRef_stop(scope,attr) ::= "retval.Stop"
rulePropertyRef_tree(scope,attr) ::= "retval.Tree"
rulePropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
input.TokenStream.ToString(
  input.TreeAdaptor.GetTokenStartIndex(retval.Start),
  input.TreeAdaptor.GetTokenStopIndex(retval.Start))
<else>
input.ToString(retval.Start,input.LT(-1))
<endif>
%>
rulePropertyRef_st(scope,attr) ::= "retval.Template"

lexerRulePropertyRef_text(scope,attr) ::= "Text"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(CharIndex-1)"
lexerRulePropertyRef_int(scope,attr) ::= "int.Parse(<scope>.Text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.Tree = <expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.Template =<expr>;"

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <%
<if(backtracking)>
if (<actions.(actionScope).synpredgate><\n>
{<\n>
<@indentedAction()><\n>
}
<else>
<action>
<endif>
%>

@execAction.indentedAction() ::= <<
<action>
>>

/** How to always execute an action even when backtracking */

```

```

execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
public static readonly BitSet <name> = new BitSet(new ulong[] { <words64: {it|<it>UL}; separator=", "> });
>>

codeFileExtension() ::= ".cs"

true_value() ::= "true"
false_value() ::= "false"

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/CSharp3.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD license"]
Copyright (c) 2005-2012 Terence Parr
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright
   notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright
   notice, this list of conditions and the following disclaimer in the
   documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products
   derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

/** Templates for building ASTs during normal parsing.
*

```

- * Deal with many combinations. Dimensions are:
- * Auto build or rewrite
- * no label, label, list label (label/no-label handled together)
- * child, root
- * token, set, rule, wildcard
- *
- * The situation is not too bad as rewrite (->) usage makes ^ and !
- * invalid. There is no huge explosion of combinations.
- */

```
finishedBacktracking(block) ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(backtracking)>
if <actions.(actionScope).synpredgate>:
  <block>
<else>
<block>
<endif>
<endif>
>>
```

```
@ruleBody.setErrorReturnValue() ::= <<
retval.tree = self._adaptor.errorNode(self.input, retval.start, self.input.LT(-1), re)
>>
```

```
// TOKEN AST STUFF
```

```
/** ID and output=AST */
tokenRef(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.tokenRef(...)>
<finishedBacktracking({
<label>_tree = <createNodeFromToken(...)>
self._adaptor.addChild(root_0, <label>_tree)
})>
>>
```

```
/** ID! and output=AST (same as plain tokenRef) */
tokenRefBang(token,label,elementIndex,terminalOptions={ }) ::= "<super.tokenRef(...)>"
```

```
/** ID^ and output=AST */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.tokenRef(...)>
<finishedBacktracking({
<label>_tree = <createNodeFromToken(...)>
root_0 = self._adaptor.becomeRoot(<label>_tree, root_0)
})>
>>
```

```

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<listLabel(elem=label,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= <%
<super.matchSet(postmatchCode={ <finishedBacktracking({self._adaptor.addChild(root_0,
<createNodeFromToken(...)>})>}, ...)>
%>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={}) ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= "<super.matchSet(...)>"

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={}) ::= <<
<if(label)>
<label> = self.input.LT(1)<n>

```

```

<endif>
<super.matchSet(postmatchCode={ <finishedBacktracking({root_0 =
self._adaptor.becomeRoot(<createNodeFromToken(...)>, root_0)})>, ...}>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<finishedBacktracking({self._adaptor.addChild(root_0, <label>.tree})>
>>

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<finishedBacktracking({root_0 = self._adaptor.becomeRoot(<label>.tree, root_0)})>
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.tree})>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabel(label, {<label>.tree})>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(label, {<label>.tree})>
>>

// WILDCARD AST

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<finishedBacktracking({
<label>_tree = self._adaptor.createWithPayload(<label>)
self._adaptor.addChild(root_0, <label>_tree)
})>

```

>>

```
wildcardBang(label,elementIndex) ::= "<super.wildcard(...)>"
```

```
wildcardRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<  
<super.wildcard(...)>  
<finishedBacktracking({  
<label>_tree = self._adaptor.createWithPayload(<label>  
root_0 = self._adaptor.becomeRoot(<label>_tree, root_0)  
})>  
>>
```

```
createNodeFromToken(label,terminalOptions={ }) ::= <%  
<if(terminalOptions.node)>  
<terminalOptions.node>(<label>) <! new MethodNode(IDLabel) !>  
<else>  
self._adaptor.createWithPayload(<label>  
<endif>  
>%
```

```
ruleCleanUp() ::= <<  
<super.ruleCleanUp()>  
<finishedBacktracking({  
retval.tree = self._adaptor.rulePostProcessing(root_0)  
self._adaptor.setTokenBoundaries(retval.tree, retval.start, retval.stop)  
})>  
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/Python3/ASTParser.stg  
No license file was found, but licenses were detected in source scan.
```

/*

[The "BSD license"]

Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC

<http://www.temporal-wave.com>

<http://www.linkedin.com/in/jimidle>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the

documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to AST stuff. Dynamic inheritance

* hierarchy is set up as ASTDbg : AST : Dbg : Java by code generator.

*/

parserMembers() ::= <<

protected DebugTreeAdaptor adaptor =

new DebugTreeAdaptor(null,new CommonTreeAdaptor());

public void setTreeAdaptor(TreeAdaptor adaptor) {

this.adaptor = new DebugTreeAdaptor(dbg,adaptor);

}

public TreeAdaptor getTreeAdaptor() {

return adaptor;

}<\n>

>>

parserCtorBody() ::= <<

>>

createListenerAndHandshake() ::= <<

<super.createListenerAndHandshake(>

>>

ctorForPredefinedListener() ::= <<

>>

@rewriteElement.pregen() ::= ""

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/C/ASTDbg.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2006 Kay Roepke 2010 Alan Condit

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/ObjC/Dbg.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2006, 2007 Kay Roepke 2010 Alan Condit

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

- * Template group file for the Objective C code generator.
- * Heavily based on Java.stg
- *
- * Written by Kay Roepke <kroepke(at)classdump.org>
- * Modified by Alan Condit <acondit(at)ipns.com>
- *
- * This file is part of ANTLR and subject to the same license as ANTLR itself.

*/

objcTypeInitMap ::= [

```
"int"      : "0",      // Integers  start out being 0
"long"     : "0",      // Longs    start out being 0
"float"    : "0.0",    // Floats   start out being 0
"double"   : "0.0",    // Doubles  start out being 0
"BOOL"     : "NO",     // Booleans start out being Antlr ObjC for false
"byte"     : "0",      // Bytes    start out being 0
"short"    : "0",      // Shorts   start out being 0
"char"     : "0",      // Chars    start out being 0
"id"       : "nil",    // ids      start out being nil
default    : "nil"     // anything other than an atomic type
```

]

// System.Boolean.ToString() returns "True" and "False", but the proper C# literals are "true" and "false"

// The Java version of Boolean returns "true" and "false", so they map to themselves here.

booleanLiteral ::= [

```
"True": "true",
"False": "false",
"true": "YES",
"false": "NO",
default: "NO"
```

]

className() ::= "<name><!<if(LEXER)>Lexer<else><if(TREE_PARSER)>Tree<endif>Parser<endif>!>"

leadIn(type) ::=

<<

```

/** \file
 * This <type> file was generated by $ANTLR version <ANTLRVersion>
 *
 * - From the grammar source file : <fileName>
 * - On : <generatedTimestamp>
<if(LEXER)>
 * - for the lexer : <name>Lexer
<endif>
<if(PARSER)>
 * - for the parser : <name>Parser
<endif>
<if(TREE_PARSER)>
 * - for the tree parser : <name>TreeParser
<endif>
 *
 * Editing it, at least manually, is not wise.
 *
 * ObjC language generator and runtime by Alan Condit, acondit|hereisanat|ipns|dotgoeshere|com.
 *
 *
 >>

/** The overall file structure of a recognizer; stores methods for rules
 * and cyclic DFAs plus support code.
 */
outputFile( LEXER,
            PARSER,
            TREE_PARSER,
            actionScope,
            actions,
            docComment,
            recognizer,
            name,
            tokens,
            tokenNames,
            rules,
            cyclicDFAs,
            bitsets,
            buildTemplate,
            buildAST,
            rewriteMode,
            profile,
            backtracking,
            synpreds,
            memoize,
            numRules,
            fileName,
            ANTLRVersion,

```

```

        generatedTimestamp,
        trace,
        scopes,
        superClass,
        literals
    ) ::=
<<
<leadIn("OBJC source")>
*/
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>

<! <if(actions.(actionScope).header)>
/* =====
* This is what the grammar programmer asked us to put at the top of every file.
*/
<actions.(actionScope).header>
/* End of Header action.
* =====
*/
<endif> !>

/* -----
* Include the ANTLR3 generated header file.
*/
#import "<name><!<if(LEXER)>Lexer<else><if(TREE_PARSER)>Tree<endif>Parser<endif>!>.h"
<actions.(actionScope).postinclude>
/* ----- */

<docComment>

<if(literals)>
/** String literals used by <name> that we must do things like MATCHS() with.
* C will normally just lay down 8 bit characters, and you can use L"xxx" to
* get wchar_t, but wchar_t is 16 bits on Windows, which is not UTF32 and so
* we perform this little trick of defining the literals as arrays of UINT32
* and passing in the address of these.
*/
<literals:{it | static ANTLR3_UCHAR lit_<i>[] = <it>;}; separator="\n">

<endif>

/* ===== */
/* ===== */
* Start of recognizer
*/
<recognizer>
>>
headerFileExtension() ::= ".h"

```

```

headerFile( LEXER,
    PARSER,
    TREE_PARSER,
    actionScope,
    actions,
    docComment,
    recognizer,
    name,
    tokens,
    tokenNames,
    rules,
    cyclicDFAs,
    bitsets,
    buildTemplate,
    buildAST,
    rewriteMode,
    profile,
    backtracking,
    synpreds,
    memoize,
    numRules,
    fileName,
    ANTLRVersion,
    generatedTimestamp,
    trace,
    scopes,
    superClass,
    literals
) ::=

<<
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>

<@imports>
<actions.(actionScope).preincludes>
/* =====
* Standard antlr OBJC runtime definitions
*/
#import \<Cocoa/Cocoa.h>
#import \<ANTLR/ANTLR.h>
/* End of standard antlr3 runtime definitions
* =====
*/
<actions.(actionScope).includes>
<@end>

<if(LEXER)>
<lexerHeaderFile(...)>

```

```

<endif>
<if(PARSER)>
<parserHeaderFile(...)>
<endif>
<if(TREE_PARSER)>
<treeParserHeaderFile(...)>
<endif>
<docComment>
>>

```

```

lexerHeaderFile( LEXER,
    PARSER,
    TREE_PARSER,
    actionScope,
    actions,
    docComment,
    recognizer,
    name,
    tokens,
    tokenNames,
    rules,
    cyclicDFAs,
    bitsets,
    buildTemplate,
    profile,
    backtracking,
    synpreds,
    memoize,
    numRules,
    fileName,
    ANTLRVersion,
    generatedTimestamp,
    trace,
    scopes,
    superClass="Lexer"
) ::=

```

```
<<
```

```

<if(actions.(actionScope).header)>
/* =====
* This is what the grammar programmer asked us to put at the top of every file.
*/
<actions.(actionScope).header>
/* End of Header action.
* =====
*/
<endif>

```

```

/* Start cyclicDFAInterface */
<cyclicDFAs:cyclicDFAInterface(>

#pragma mark Rule return scopes Interface start
<rules:{rule |
<rule.ruleDescriptor:{ruleDescriptor | <returnScopeInterface(scope=ruleDescriptor.returnScope)>>>>
#pragma mark Rule return scopes Interface end
#pragma mark Tokens
#ifndef TOKENLISTAlreadyDefined
#define TOKENLISTAlreadyDefined 1
#ifdef EOF
#undef EOF
#endif
<tokens:{it | #define <it.name> <it.type>}; separator="\n">
#endif
/* interface lexer class */
@interface <className()> <@superClassName>: <superClass><@end> { // line 283
<cyclicDFAs:{dfa | DFA<dfa.decisionNumber> *dfa<dfa.decisionNumber>;}; separator="\n">
<synpreds:{pred | SEL <pred>Selector;}; separator="\n">
/* ObjC start of actions.lexer.memVars */
<actions.lexer.memVars>
/* ObjC end of actions.lexer.memVars */
}
+ (void) initialize;
+ (<className()> *)new<className()>WithCharStream:(id<CharStream>)anInput;
/* ObjC start actions.lexer.methodsDecl */
<actions.lexer.methodsDecl>
/* ObjC end actions.lexer.methodsDecl */
<rules:{rule |
- (<rule.ruleDescriptor:{ruleDescriptor|<returnType()>>>)
<if(!rule.ruleDescriptor.isSynPred)>m<rule.ruleName><else><rule.ruleName>_fragment<endif>
<if(rule.ruleDescriptor.parameterScope)><rule.ruleDescriptor.parameterScope:parameterScope()><endif>; };
separator="\n"><\n>
@end /* end of <className()> interface */<\n>
>>

headerReturnScope(ruleDescriptor) ::= "<returnScopeInterface(...)>"
headerReturnType(ruleDescriptor) ::= <<
<if(LEXER)>
<if(!r.ruleDescriptor.isSynPred)>
void
<else>
<ruleDescriptor:returnType()>
<endif>
<else>
<ruleDescriptor:returnType()>
<endif>
>>

```

```

// Produce the lexer output
lexer( grammar,
    name,
    tokens,
    scopes,
    rules,
    numRules,
    filterMode,
    labelType="CommonToken",
    superClass="Lexer"
) ::= <<
<cyclicDFAs:cyclicDFA(>

/** As per Terence: No returns for lexer rules! */
<!
#pragma mark Rule return scopes start
<rules:{rule | <rule.ruleDescriptor:{ruleDescriptor |
<returnScopeImplementation(scope=ruleDescriptor.returnScope)>>
}>
#pragma mark Rule return scopes end
!>
@implementation <grammar.recognizerName> // line 330

+ (void) initialize
{
    [BaseRecognizer setGrammarFileName:@"<fileName>"];
}

+ (NSString *) tokenNameForType:(NSInteger)aTokenType
{
    return [[self getTokenNames] objectAtIndex:aTokenType];
}

+ (<grammar.recognizerName> *)new<grammar.recognizerName>WithCharStream:(id<CharStream>)anInput
{
    return [[<grammar.recognizerName> alloc] initWithCharStream:anInput];
}

- (id) initWithCharStream:(id<CharStream>)anInput
{
    self = [super initWithCharStream:anInput state:[RecognizerSharedState
newRecognizerSharedStateWithRuleLen:<numRules>+1]];
    if ( self != nil ) {
<if(memoize)>
        if ( state.ruleMemo == nil ) {
            state.ruleMemo = [[RuleStack newRuleStackWithSize:<numRules>+1] retain];
        }
        if ( [state.ruleMemo count] == 0 ) {

```

```

        // initialize the memoization cache - the indices are 1-based in the runtime code!
        <! [state.ruleMemo addObject:[NSNull null]]; /* dummy entry to ensure 1-basedness. */ !>
        for (NSInteger i = 0; i \< <numRules>; i++) {
            [state.ruleMemo addObject:[HashRule newHashRuleWithLen:17]];
        }
    }
<endif>
    <synpreds:{ pred | <lexerSynpred(name=pred)>};separator="\n">
    <cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = [DFA<dfa.decisionNumber>
newDFA<dfa.decisionNumber>WithRecognizer:self];}; separator="\n">
    <actions.lexer.init>
    }
    return self;
}

- (void) dealloc
{
    <cyclicDFAs:{ dfa | [dfa<dfa.decisionNumber> release];}; separator="\n">
<actions.lexer.dealloc>
    [super dealloc];
}

/* ObjC Start of actions.lexer.methods */
<actions.lexer.methods>
/* ObjC end of actions.lexer.methods */
/* ObjC start methods() */
<@methods()>
/* ObjC end methods() */

<if(actions.lexer.reset)>
- (void) reset
{
    <actions.lexer.reset>
    [super reset];
}
<endif>

<if(filterMode)>
<filteringNextToken()>
<endif>
/* Start of Rules */
<rules; separator="\n">

@end /* end of <grammar.recognizerName> implementation line 397 */
>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume

```



```

* a token and then try again. backtracking needs to be set as well.
* Make rule memoization happen only at levels above 1 as we start mTokens
* at backtracking==1.
*/
filteringNextToken() ::= <<
- (id\<Token>) nextToken
{
  while (YES) {
    if ( [input LA:1] == CharStreamEOF ) {
      return [<labelType> eofToken];
    }
    state.token = nil;
    state.channel = TokenChannelDefault;
    state.tokenStartCharIndex = input.index;
    state.tokenStartCharPositionInLine = input.charPositionInLine;
    state.tokenStartLine = input.line;
    state.text = nil;
    @try {
      NSInteger m = [input mark];
      state.backtracking = 1; /* means we won't throw slow exception */
      state.failed = NO;
      [self mTokens];
      state.backtracking = 0;
      /* mTokens backtracks with synpred at backtracking==2
      and we set the synpredgate to allow actions at level 1. */
      if ( state.failed ) {
        [input rewind:m];
        [input consume]; /* advance one char and try again */
      } else {
        [self emit];
        return state.token;
      }
    }
    @catch (RecognitionException *re) {
      // shouldn't happen in backtracking mode, but...
      [self reportError:re];
      [self recover:re];
    }
  }
}

- (void)memoize:(id\<IntStream\>)anInput
  RuleIndex:(NSInteger)ruleIndex
  StartIndex:(NSInteger)ruleStartIndex
{
  if ( state.backtracking > 1 ) [super memoize:anInput RuleIndex:ruleIndex StartIndex:ruleStartIndex];
}

```

```

- (BOOL)alreadyParsedRule:(id)<IntStream>anInput RuleIndex:(NSInteger)ruleIndex
{
    if ( state.backtracking > 1 ) return [super alreadyParsedRule:anInput RuleIndex:ruleIndex];
    return NO;
}
>>

```

```

actionGate() ::= "state.backtracking == 0"

```

```

filteringActionGate() ::= "state.backtracking == 1"

```

```

parserHeaderFile( LEXER,
    PARSER,
    TREE_PARSER,
    actionScope,
    actions,
    docComment,
    recognizer,
    name,
    tokens,
    tokenNames,
    rules,
    cyclicDFAs,
    bitsets,
    buildTemplate,
    profile,
    backtracking,
    synpreds,
    memoize,
    numRules,
    fileName,
    ANTLRVersion,
    generatedTimestamp,
    trace,
    scopes,
    literals,
    superClass="Parser"
) ::= <<
/* parserHeaderFile */
<genericParserHeaderFile(inputStreamType="id<TokenStream>",...)>
>>

```

```

treeParserHeaderFile( LEXER,
    PARSER,
    TREE_PARSER,
    actionScope,
    actions,
    docComment,

```

```

recognizer,
name,
tokens,
tokenNames,
rules,
cyclicDFAs,
bitsets,
buildTemplate,
profile,
backtracking,
synpreds,
memoize,
numRules,
fileName,
ANTLRVersion,
generatedTimestamp,
trace,
scopes,
literals,
superClass="TreeParser"
) ::= <<
/* treeParserHeaderFile */
<genericParserHeaderFile(inputStreamType="id\<TreeNodeStream>",...)>
>>

```

```

genericParserHeaderFile( LEXER,
    PARSE,
    TREE_PARSER,
    actionScope,
    actions,
    docComment,
    recognizer,
    name,
    tokens,
    tokenNames,
    rules,
    cyclicDFAs,
    bitsets,
    buildTemplate,
    profile,
    backtracking,
    synpreds,
    memoize,
    numRules,
    fileName,
    ANTLRVersion,
    generatedTimestamp,
    trace,

```

```

        scopes,
        superClass,
        literals,
        inputStreamType
    ) ::=
<<
<if(actions.(actionScope).header)>
/* =====
* This is what the grammar programmer asked us to put at the top of every file.
*/
<actions.(actionScope).header>
/* End of Header action.
* =====
*/
<endif>

#ifndef ANTLR3TokenTypeAlreadyDefined
#define ANTLR3TokenTypeAlreadyDefined
typedef enum {
    ANTLR_EOF = -1,
    INVALID,
    EOR,
    DOWN,
    UP,
    MIN
} ANTLR3TokenType;
#endif

<cyclicDFAs:cyclicDFAInterface(>
#pragma mark Tokens
#ifndef TOKENLISTAlreadyDefined
#define TOKENLISTAlreadyDefined 1
#ifdef EOF
#undef EOF
#endif
#endif
<tokens:{it | #define <it.name> <it.type>}; separator="\n">
#endif
#pragma mark Dynamic Global Scopes globalAttributeScopeInterface
<scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeInterface(scope=it)><endif>}>
#pragma mark Dynamic Rule Scopes ruleAttributeScopeInterface
<rules:{rule |
<rule.ruleDescriptor:{ ruleDescriptor | <ruleAttributeScopeInterface(scope=ruleDescriptor.ruleScope)>}>>
#pragma mark Rule Return Scopes returnScopeInterface
<rules:{rule |<rule.ruleDescriptor:{ ruleDescriptor |
<returnScopeInterface(scope=ruleDescriptor.returnScope)>}>>

/* Interface grammar class */
@interface <className()> <@superClassName> : <superClass><@end> { /* line 572 */

```

```

#pragma mark Dynamic Rule Scopes ruleAttributeScopeDecl
<rules:{ rule | <rule.ruleDescriptor.ruleScope:ruleAttributeScopeDecl(scope=rule.ruleDescriptor.ruleScope)>>}>
#pragma mark Dynamic Global Rule Scopes globalAttributeScopeMemVar
<scopes:{ it | <if(it.isDynamicGlobalScope)><globalAttributeScopeMemVar(scope=it)><endif>><\n>
/* ObjC start of actions.(actionScope).memVars */
<actions.(actionScope).memVars>
/* ObjC end of actions.(actionScope).memVars */
/* ObjC start of memVars */
<@memVars()>
/* ObjC end of memVars */

<cyclicDFAs:{ dfa | DFA<dfa.decisionNumber> *dfa<dfa.decisionNumber>; }; separator="\n">
<synpreds:{ pred | SEL <pred>Selector; }; separator="\n">
}

/* ObjC start of actions.(actionScope).properties */
<actions.(actionScope).properties>
/* ObjC end of actions.(actionScope).properties */
/* ObjC start of properties */
<@properties()>
/* ObjC end of properties */

+ (void) initialize;
+ (<className()> *) new<className()>:(<inputStreamType>)aStream;
/* ObjC start of actions.(actionScope).methodsDecl */
<actions.(actionScope).methodsDecl>
/* ObjC end of actions.(actionScope).methodsDecl */

/* ObjC start of methodsDecl */
<@methodsDecl()>
/* ObjC end of methodsDecl */

<rules:{ rule |
-
(<rule.ruleDescriptor:{ ruleDescriptor|<returnType()>>)<if(!rule.ruleDescriptor.isSynPred)><rule.ruleName><else
><rule.ruleName>_fragment<endif><if(rule.ruleDescriptor.parameterScope)><rule.ruleDescriptor.parameterScope:
parameterScope()><endif>; }; separator="\n"><\n>

@end /* end of <className()> interface */<\n>
>>

parser( grammar,
    name,
    scopes,
    tokens,
    tokenNames,
    rules,
    numRules,

```

```

    bitsets,
    ASTLabelType="CommonTree",
    superClass="Parser",
    labelType="CommonToken",
    members={<actions.parser.members>}
  ) ::= <<
<genericParser(inputStreamType="id\<TokenStream>", rewriteElementType="Token", ...)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser( grammar,
  name,
  scopes,
  tokens,
  tokenNames,
  globalAction,
  rules,
  numRules,
  bitsets,
  filterMode,
  labelType={<ASTLabelType>},
  ASTLabelType="CommonTree",
  superClass={<if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif>},
  members={<actions.treeparser.members>}
) ::= <<
<genericParser(inputStreamType="id\<TreeNodeStream>", rewriteElementType="Node", ...)>
>>

/** How to generate a parser */
genericParser( grammar,
  name,
  scopes,
  tokens,
  tokenNames,
  rules,
  numRules,
  cyclicDFAs,      // parser init -- initializes the DFAs
  bitsets,
  labelType,
  ASTLabelType,
  superClass,
  members,
  filterMode,
  rewriteElementType,
  inputStreamType
) ::= <<

```

```

<cyclicDFAs:cyclicDFA()>

#pragma mark Bitsets
<bitsets:{it | <bitset(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>}, words64=it.bits)>>>

#pragma mark Dynamic Global globalAttributeScopeImplementation
<scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeImplementation(scope=it)><endif>>>

#pragma mark Dynamic Rule Scopes ruleAttributeScopeImplementation
<rules:{rule |
<rule.ruleDescriptor:{ ruleDescriptor | <ruleAttributeScopeImplementation(scope=ruleDescriptor.ruleScope)>>>>

#pragma mark Rule Return Scopes returnScopeImplementation
<rules:{rule | <rule.ruleDescriptor:{ ruleDescriptor |
<returnScopeImplementation(scope=ruleDescriptor.returnScope)>>>>

@implementation <grammar.recognizerName> // line 637

/* ObjC start of ruleAttributeScope */
#pragma mark Dynamic Rule Scopes ruleAttributeScope
<rules:{rule | <rule.ruleDescriptor.ruleScope:ruleAttributeScope()>>>
/* ObjC end of ruleAttributeScope */
#pragma mark global Attribute Scopes globalAttributeScope
/* ObjC start globalAttributeScope */
<scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScope()><endif>>>
/* ObjC end globalAttributeScope */
/* ObjC start actions.(actionScope).synthesize */
<actions.(actionScope).synthesize>
/* ObjC start synthesize() */
<@synthesize()>

+ (void) initialize
{
    #pragma mark Bitsets
    <bitsets:{it | <bitsetInit(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>}, words64=it.bits)>>>
    [BaseRecognizer setTokenNames:[[AMutableArray arrayWithObjects:@"\<invalid>", @"\<EOR>",
@"\<DOWN>", @"\<UP>", <tokenNames:{it | @<it>}; separator=",", wrap="\n", nil] retain]];
    [BaseRecognizer setGrammarFileName:@"<fileName>"];
    <synpreds:{pred | <synpred(pred)>>>
}

+ (<grammar.recognizerName> *)new<grammar.recognizerName>:(<inputStreamType>)aStream
{
<if(PARSER)>
    return [[<grammar.recognizerName> alloc] initWithTokenStream:aStream];
<else><! TREE_PARSER !>
    return [[<grammar.recognizerName> alloc] initWithStream:aStream];
<endif>

```

```

}

<if(PARSER)>
- (id) initWithTokenStream:(<inputStreamType>)aStream
{
    self = [super initWithTokenStream:aStream State:[[RecognizerSharedState
newRecognizerSharedStateWithRuleLen:<numRules>+1] retain]];
    if ( self != nil ) {
<else><! TREE_PARSER !>
- (id) initWithStream:(<inputStreamType>)aStream
{
    self = [super initWithStream:aStream State:[[RecognizerSharedState
newRecognizerSharedStateWithRuleLen:<numRules>+1] retain]];
    if ( self != nil ) {
<endif>
        <! <parserCtorBody()> !>
        <cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = [DFA<dfa.decisionNumber>
newDFA<dfa.decisionNumber>WithRecognizer:self];}; separator="\n">
        <scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeInit(scope=it)><endif>}>
        <rules:{rule | <rule.ruleDescriptor.ruleScope:ruleAttributeScopeInit()>>
        /* start of actions-actionScope-init */
        <actions.(actionScope).init>
        /* start of init */
        <@init()>
    }
    return self;
}

- (void) dealloc
{
    <cyclicDFAs:{ dfa | [dfa<dfa.decisionNumber> release];}; separator="\n">
    <scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeDealloc(scope=it)><endif>}>
    <actions.(actionScope).dealloc>
    <@dealloc()>
    [super dealloc];
}

/* ObjC start actions.(actionScope).methods */
<actions.(actionScope).methods>
/* ObjC end actions.(actionScope).methods */
/* ObjC start methods() */
<@methods()>
/* ObjC end methods() */
/* ObjC start rules */
<rules; separator="\n">
/* ObjC end rules */

@end /* end of <grammar.recognizerName> implementation line 692 */<\n>

```


>>

```
parserCtorBody() ::= <<
<if(memoize)> /* parserCtorBody */
<if(grammar.grammarIsRoot)>
state.ruleMemo = [[RuleStack newRuleStack:<numRules>+1] retain];<\n> <! index from 1..n !>
<endif>
<endif>
<grammar.delegators:
{g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
>>
```

```
/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
```

```
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start <ruleName>_fragment
- (void) <ruleName>_fragment
{
  <ruleLabelDefs()>
  <if(trace)>
    [self traceIn:@"<ruleName>_fragment" Index:<ruleDescriptor.index>];
    @try {
      <block>
    }
    @finally {
      [self traceOut:@"<ruleName>_fragment" Index:<ruleDescriptor.index>];
    }
  <else>
    <block>
  <endif>
} // $ANTLR end <ruleName>_fragment
>>
```

```
synpred(name) ::= <<
SEL <name>Selector = @selector(<name>_fragment);
<! // $ANTLR start <name>
- (BOOL) <name>
{
  state.backtracking++;
  <@start()>
  NSInteger start = [input mark];
  @try {
```

```

    [self <name>_fragment]; // can never throw exception
}
@catch (RecognitionException *re) {
    NSLog(@"impossible: %@\n", re.name);
}
BOOL success = (state.failed == NO);
[input rewind:start];
<@stop()>
state.backtracking--;
state.failed=NO;
return success;
} // $ANTLR end <name> <\n !>
>>

lexerSynpred(name) ::= <<
<synpred(name)>
>>

ruleMemoization(name) ::= <<
<if(memoize)>
if ( state.backtracking > 0 && [self alreadyParsedRule:input RuleIndex:<ruleDescriptor.index>] ) { return
<ruleReturnValue()>; }
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if (backtracking)>if ( state.failed ) return <ruleReturnValue()>;<endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if ( state.backtracking > 0 ) { state.failed = YES; return <ruleReturnValue()>; }<\n><endif>
>>

/** How to generate code for a rule.
* The return type aggregates are declared in the header file (headerFile template)
*/
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<

/*
* $ANTLR start <ruleName>
* <fileName>:<description>
*/
- (<returnType()>) <ruleName><ruleDescriptor.parameterScope:parameterScope()>
{
    <if(trace)>[self traceIn:\@"<ruleName>" Index:<ruleDescriptor.index>];<endif>
    <if(trace)>NSLog(@"enter <ruleName> %@ failed=%@ backtracking=%d", [input LT:1],

```

```

(state.failed==YES)?@"YES":@"NO", state.backtracking);<endif>
  <ruleScopeSetUp()>
  <ruleDeclarations()>
  <ruleDescriptor.actions.init>
  <@preamble()>
  @try {
    <ruleMemoization(name=ruleName)>
    <ruleLabelDefs()>
    <block>
    <ruleCleanup()>
    <(ruleDescriptor.actions.after):execAction()>
  }
  <if(exceptions)>
    <exceptions: {e|<catch(decl=e.decl,action=e.action)><\n}>>
  <else><if(!emptyRule)><if(actions.(actionScope).rulecatch)>
    <actions.(actionScope).rulecatch>
  <else>
    @catch (RecognitionException *re) {
      [self reportError:re];
      [self recover:input Exception:re];
      <@setErrorReturnValue()>
    }<\n>
  <endif><endif><endif>
  @finally {
    <if(trace)>[self traceOut:@"<ruleName>" Index:<ruleDescriptor.index>];<endif>
    <memoize()>
    <ruleScopeCleanup()>
    <finally>
  }
  <@postamble()>
  return <ruleReturnValue()>;
}
/* $ANTLR end <ruleName> */
>>

finalCode(finalBlock) ::= <<
{
  <finalBlock>
}
>>

catch(decl,action) ::= <<
@catch (<e.decl>) {
  <e.action>
}
>>

ruleDeclarations() ::= <<

```

```

/* ruleDeclarations */
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType()> retval = [<ruleDescriptor:returnStructName()> new<ruleDescriptor:returnStructName()>];
[retval setStart:[input LT:1]];<\n>
<else>
<ruleDescriptor.returnScope.attributes:{ a |
<a.type> <a.name> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
NSInteger <ruleDescriptor.name>_startIndex = input.index;
<endif>
>>

ruleScopeSetUp() ::= <<
/* ruleScopeSetUp */
<ruleDescriptor.useScopes:{it | [<it>_stack push:[<it>_Scope new<it>_Scope]];}>
<ruleDescriptor.ruleScope:{it | [<it.name>_stack push:[<it.name>_Scope new<it.name>_Scope]];}>
>>

ruleScopeCleanUp() ::= <<
/* ruleScopeCleanUp */
<ruleDescriptor.useScopes:{it | [<it>_stack pop];}; separator="\n">
<ruleDescriptor.ruleScope:{it | [<it.name>_stack pop];}; separator="\n">
>>

ruleLabelDefs() ::= <%
/* ruleLabelDefs entry */<\n">
<[ruleDescriptor.tokenLabels, ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it | <labelType> *<it.label.text> = nil;}; separator="\n"><\n">
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: {it | AMutableArray *list_<it.label.text> = nil;}; separator="\n"><\n">
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n"><\n">
<ruleDescriptor.ruleListLabels:{||ParserRuleReturnScope *<it.label.text> = nil;}; separator="\n"><\n">
%>

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | <labelType> *<it.label.text>=nil;}; separator="\n"
>
<ruleDescriptor.charLabels:{it |NSInteger <it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]:{it |AMutableArray *list_<it.label.text>=nil; }; separator="\n">
>>

```

```

ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
%>

ruleCleanup() ::= <<
/* token+rule list labels */
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]:{it |[list_<it.label.text> release];}; separator="\n">
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
[retval setStop:[input LT:-1]];<\n>
<endif><endif>
>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if (state.backtracking > 0) [self memoize:input RuleIndex:<ruleDescriptor.index>
StartIndex:<ruleDescriptor.name>_StartIndex];
<endif><endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName, nakedBlock, ruleDescriptor, block, memoize) ::= <<
// $ANTLR start "<ruleName>"
- (void)
m<ruleName><if(ruleDescriptor.parameterScope)><ruleDescriptor.parameterScope:parameterScope(scope=it)><en
dif>
{
//<if(trace)>[self traceIn:\@"<ruleName>" Index:<ruleDescriptor.index>];<endif>
<if(trace)>NSLog(@"enter <ruleName> %C line=%d:%d failed=%@ backtracking=%d",
[input LA:1],
self.line,
self.charPositionInLine,
(state.failed==YES) ? @"YES" : @"NO",
state.backtracking);
<endif>
<ruleScopeSetUp()>
<ruleDeclarations()>

```

```

    @try {
<if(nakedBlock)>
    <ruleMemoization(name=ruleName)>
    <lexerRuleLabelDefs()>
    <ruleDescriptor.actions.init>
    <block><\n>
<else>
    NSInteger _type = <ruleName>;
    NSInteger _channel = TokenChannelDefault;
    <ruleMemoization(name=ruleName)>
    <lexerRuleLabelDefs()>
    <ruleDescriptor.actions.init>
    <block>
    <ruleCleanUp()>
    state.type = _type;
    state.channel = _channel;
    <(ruleDescriptor.actions.after):execAction()>
<endif>
    }
    @finally {
        //<if(trace)>[self traceOut:[NSString stringWithFormat:@"<ruleName> %d\n",
<ruleDescriptor.index>]];<endif>
        <if(trace)>NSLog(@"exit <ruleName> %C line=%d:%d failed=%@ backtracking=%d",
            [input LA:1], self.line, self.charPositionInLine,
            (state.failed==YES) ? @"YES" : @"NO", state.backtracking);<endif>
        <ruleScopeCleanUp()>
        <memoize()>
    }
    return;
}
/* $ANTLR end "<ruleName>" */
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
- (void) mTokens
{
    <block><\n>
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description> // block

```

```

NSInteger alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
<@prebranch()>
switch (alt<decisionNumber>) {
  <alts:{ a | <altSwitchCase(i, a)>}>
}
<@postbranch()>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description> //ruleblock
NSInteger alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
switch (alt<decisionNumber>) {
  <alts:{ a | <altSwitchCase(i, a)>}>
}
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description> // ruleBlockSingleAlt
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description> // blockSingleAlt
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description> // positiveClosureBlock
NSInteger cnt<decisionNumber> = 0;
<decls>

```

```

<@preloop()>
do {
  NSInteger alt<decisionNumber> = <maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  switch (alt<decisionNumber>) {
    <alts:{ a | <altSwitchCase(i, a)>}>
    default :
      if ( cnt<decisionNumber> >= 1 )
        goto loop<decisionNumber>;
      <ruleBacktrackFailure()>
      EarlyExitException *eee =
        [EarlyExitException newException:input decisionNumber:<decisionNumber>];
      <@earlyExitException()>
      @throw eee;
  }
  cnt<decisionNumber>++;
} while (YES);
loop<decisionNumber>;
<@postloop()>
>>

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

*/** A (..)* block with 0 or more alternatives */*

closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=

<<

<decls>

<@preloop()>

do {

NSInteger alt<decisionNumber>=<maxAlt>;

<@predecision()>

<decision>

<@postdecision()>

switch (alt<decisionNumber>) {

<alts:{ a | <altSwitchCase(i, a)>}>

default :

goto loop<decisionNumber>;

}

} while (YES);

loop<decisionNumber>;

<@postloop()>

>>

closureBlockSingleAlt ::= closureBlock

*/** Optional blocks (x)? are translated to (x|) by before code generation*


```

* so we can just use the normal block template
*/
optionalBlock ::= block

optionalBlockSingleAlt ::= block

/** A case in a switch that jumps to an alternative given the alternative
* number. A DFA predicts the alternative and then a simple switch
* does the jump to the code that actually matches that alternative.
*/
altSwitchCase(altNum, alt) ::= <<
case <altNum> : ;
  <@prealt(>
  <alt>
  break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description> // alt
{
<@declarations(>
<elements:element(>
<rew>
<@cleanup(>
}
>>

/** What to emit when there is no rewrite. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(e) ::= << <@prematch(><\n><e.el><\n> >>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(label)><label>=(<labelType> *)<endif>[self match:input TokenType:<token>
Follow:FOLLOW_<token>_in_<ruleName><elementIndex>]; <checkRuleBacktrackFailure(>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>

```

>>

```
listLabel(label,elem) ::= <<
if (list_<label> == nil) list_<label> = [[AMutableArray arrayWithCapacity:5] retain];
[list_<label> addObject:<elem>];<\n>
>>
```

```
/** match a character */
charRef(char,label) ::= <<
<if(label)>NSInteger <label> = [input LA:1];<\n><endif>
[self matchChar:<char>]; <checkRuleBacktrackFailure()><\n>
>>
```

```
/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)><label> = [input LA:1];<\n><endif>
[self matchRangeFromChar:<a> to:<b>]; <checkRuleBacktrackFailure()>
>>
```

```
/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,terminalOptions,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label> = [input LA:1];<\n>
<else>
<label> = (<labelType> *)[input LT:1]; /* matchSet */<\n>
<endif><endif>
if (<s>) {
    [input consume];
    <postmatchCode>
<if(!LEXER)>
    [state setIsErrorRecovery:NO];
<endif>
    <if(backtracking)>state.failed = NO;<\n><endif>
} else {
    <ruleBacktrackFailure()>
    MismatchedSetException *mse = [MismatchedSetException newException:nil stream:input];
    <@mismatchedSetException()>
<if(LEXER)>
<if(label)>
    mse.c = <label>;
<endif>
    [self recover:mse];
    @throw mse;
<else>
    @throw mse;
    <! use following code to make it recover inline; remove throw mse;
    [self recoverFromMismatchedSet:input exception:mse follow:FOLLOW_set_in_<ruleName><elementIndex>]; !>
```

```

<endif>
}<\n>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex="0") ::= <<
<if(label)>
NSInteger <label>Start = input.index;
[self matchString:<string>]; <checkRuleBacktrackFailure()>
NSInteger StartLine<elementIndex> = self.line;
NSInteger <label>StartCharPos<elementIndex> = self.charPositionInLine;
<label> = [[<labelType> newToken:input Type:TokenTypeInvalid Channel:TokenChannelDefault
Start:<label>Start Stop:input.index] retain];
[self setLine:<label>StartLine<elementIndex>];
[self setCharPositionInLine:<label>StartCharPos<elementIndex>];
<else>
[self matchString:<string>]; <checkRuleBacktrackFailure()><\n>
<endif>
>>

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>
<label> = (<labelType> *)[input LT:1];<\n>
<endif>
[self matchAny:input]; <checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
NSInteger <label> = [input LA:1];<\n>
<endif>
[self matchAny]; <checkRuleBacktrackFailure()><\n>
>>

wildcardCharListLabel(label, elementIndex) ::= <<

```

```

<wildcardChar(...)>
<listLabel(elem=label,...)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
/* ruleRef */
[self pushFollow:FOLLOW_<rule.name>_in_<ruleName><elementIndex>];
<if(label)><label> = <endif>[self <if(scope)><scope.delegateName()>.<endif><rule.name><if(args)>:<first(args)>
<rest(args):{ a | arg<i>:<rest(args)>} ; separator=" "><endif>];<\n>
[self popFollow];
<checkRuleBacktrackFailure()><\n>
>>

/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
NSInteger <label>Start<elementIndex> = input.index;
[self m<rule.name><if(args)>:<args; separator=":"><endif>]; <checkRuleBacktrackFailure()><\n>
<label> = [[<labelType> newToken:input Type:TokenTypeInvalid Channel:TokenChannelDefault
Start:<label>Start<elementIndex> Stop:input.index-1] retain];
<label>.line = self.line;
<else>
[self <if(scope)><scope.delegateName()>.<endif>m<rule.name><if(args)>:<args; separator=":"><endif>];
<checkRuleBacktrackFailure()><\n>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>

```

```

>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
NSInteger <label>Start<elementIndex> = input.index;
[self matchChar:CharStreamEOF]; <checkRuleBacktrackFailure()><\n>
<labelType> <label> = [[<labelType> newToken:input Type:TokenTypeEOF Channel:TokenChannelDefault
Start:<label>Start<elementIndex> Stop:input.index-1] retain];
<label>.line = self.line;
<else>
[self matchChar:CharStreamEOF]; <checkRuleBacktrackFailure()><\n>
<endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg()>"
recRuleArg()            ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList, enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( [input LA:1] == DOWN ) {
[self match:input TokenType:DOWN Follow:nil]; <checkRuleBacktrackFailure()>
<children:element()>
[self match:input TokenType:UP Follow:nil]; <checkRuleBacktrackFailure()>
}
<else>
[self match:input TokenType:DOWN Follow:nil]; <checkRuleBacktrackFailure()>
<children:element()>
[self match:input TokenType:UP Follow:nil]; <checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
* also hoisted into a prediction expression).
*/
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(...)> ) ) {
<ruleBacktrackFailure()>
@throw [FailedPredicateException newException:@ "<ruleName>" predicate:@ "<description>" stream:input];
}
>>

```

```

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
NSInteger LA<decisionNumber>_<stateNumber> = [input LA:<k>];<\n>
<edges; separator="\nelse ">
else {
<if(eotPredictsAlt)>
  alt<decisionNumber> = <eotPredictsAlt>;
<else>
  <ruleBacktrackFailure()>
  NoViableAltException *nvae = [NoViableAltException newException:<decisionNumber> state:<stateNumber>
stream:input];
  nvae.c = LA<decisionNumber>_<stateNumber>;
  <@noViableAltException()>
  @throw nvae;<\n>
<endif>
}
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
NSInteger LA<decisionNumber>_<stateNumber> = [input LA:<k>];<\n>
<edges; separator="\nelse ">
>>

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
NSInteger LA<decisionNumber>_<stateNumber> = [input LA:<k>];
<edges; separator="\nelse "><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber>=<eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else {
  alt<decisionNumber> = <eotPredictsAlt>;
}<\n>
<endif><endif>
>>

```

```

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>";

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif>) {
    <targetState>
}
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
unichar charLA<decisionNumber> = [input LA:<k>];
switch (charLA<decisionNumber>) {
    <edges; separator="\n"><\n>
default: ;
<if(eotPredictsAlt)>
    alt<decisionNumber> = <eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>
    NoViableAltException *nvae = [NoViableAltException newException:<decisionNumber> state:<stateNumber>
stream:input];
    nvae.c = charLA<decisionNumber>;
    <@noViableAltException()>
    @throw nvae;<\n>
<endif>
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ([input LA:<k>]) { // dfaOptionalBlockStateSwitch
    <edges; separator="\n"><\n>
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ([input LA:<k>]) { // dfaLoopbackStateSwitch
    <edges; separator="\n"><\n>
<if(eotPredictsAlt)>
default:

```

```

    alt<decisionNumber> = <eotPredictsAlt>;
    break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels:{it | case <it>: ;}; separator="\n">
{
    <targetState>
}
break;
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = [dfa<decisionNumber> predict:input];
>>

/** Used in headerFile */
cyclicDFAInterface(dfa) ::= <<
#pragma mark Cyclic DFA interface start DFA<dfa.decisionNumber>
@interface DFA<dfa.decisionNumber> : DFA {
}
+ (DFA<dfa.decisionNumber> *) newDFA<dfa.decisionNumber>WithRecognizer:(BaseRecognizer
*)theRecognizer;
- initWithRecognizer:(BaseRecognizer *)recognizer;
@end /* end of DFA<dfa.decisionNumber> interface */<\n>
#pragma mark Cyclic DFA interface end DFA<dfa.decisionNumber><\n>
>>

/** Used in lexer/parser implementation files */
/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
#pragma mark Cyclic DFA implementation start DFA<dfa.decisionNumber>

```



```

@implementation DFA<dfa.decisionNumber>
const static NSInteger dfa<dfa.decisionNumber>_eot[<dfa.numberofStates>] =
    {<dfa.eot; wrap="\n    ", separator=",", null="-1">};
const static NSInteger dfa<dfa.decisionNumber>_eof[<dfa.numberofStates>] =
    {<dfa.eof; wrap="\n    ", separator=",", null="-1">};
const static unichar dfa<dfa.decisionNumber>_min[<dfa.numberofStates>] =
    {<dfa.min; wrap="\n    ", separator=",", null="-1">};
const static unichar dfa<dfa.decisionNumber>_max[<dfa.numberofStates>] =
    {<dfa.max; wrap="\n    ", separator=",", null="-1">};
const static NSInteger dfa<dfa.decisionNumber>_accept[<dfa.numberofStates>] =
    {<dfa.accept; wrap="\n    ", separator=",", null="-1">};
const static NSInteger dfa<dfa.decisionNumber>_special[<dfa.numberofStates>] =
    {<dfa.special; wrap="\n    ", separator=",", null="-1">};

/** Used when there is no transition table entry for a particular state */
#define dfa<dfa.decisionNumber>_T_empty    nil

<dfa.edgeTransitionClassMap.keys:{ table |
const static NSInteger dfa<dfa.decisionNumber>_T<i0>[] =
{
    <table; separator=", ", wrap="\n ", null="-1">
\};
}; null="">

const static NSInteger *dfa<dfa.decisionNumber>_transition[] =
{
    <dfa.transitionEdgeTables:{ whichTable|dfa<dfa.decisionNumber>_T<whichTable>}; separator=", ", wrap="\n",
null="nil">
};

+ (DFA<dfa.decisionNumber> *) newDFA<dfa.decisionNumber>WithRecognizer:(BaseRecognizer *)aRecognizer
{
    return [[[DFA<dfa.decisionNumber> alloc] initWithRecognizer:aRecognizer] retain];
}

- (id) initWithRecognizer:(BaseRecognizer *) theRecognizer
{
    self = [super initWithRecognizer:theRecognizer];
    if ( self != nil ) {
        decisionNumber = <dfa.decisionNumber>;
        eot = dfa<dfa.decisionNumber>_eot;
        eof = dfa<dfa.decisionNumber>_eof;
        min = dfa<dfa.decisionNumber>_min;
        max = dfa<dfa.decisionNumber>_max;
        accept = dfa<dfa.decisionNumber>_accept;
        special = dfa<dfa.decisionNumber>_special;
        transition = dfa<dfa.decisionNumber>_transition;
    }
}

```

```

    return self;
}

<if(dfa.specialStateSTs)>
/* start dfa.specialStateSTs */
- (NSInteger) specialStateTransition:(NSInteger)s Stream:(id<IntStream>)anInput
{
<if(LEXER)>
    id<IntStream> input = anInput;<\n>
<endif>
<if(PARSER)>
    id<TokenStream> input = (id<TokenStream>)anInput;<\n>
<endif>
<if(TREE_PARSER)>
    id<TreeNodeStream> input = (id<TreeNodeStream>)anInput;<\n>
<endif>
    switch (s) {
        <dfa.specialStateSTs:{state |
            case <i0> : ;<! compressed special state numbers 0..n-1 !>
                <state>}; separator="\n">
    }
<if(backtracking)>
    if ( [recognizer getBacktrackingLevel] > 0 ) { [recognizer setFailed:YES]; return -1; }<\n>
<endif>
    NoViableAltException *nvae = [NoViableAltException newException:<dfa.decisionNumber> state:s
stream:recognizer.input];
    // nvae.c = s;
    /* [self error:nvae]; */ <! for debugger - do later !>
    @throw nvae;
}<\n>
/* end dfa.specialStateSTs */
<endif>

- (void) dealloc
{
    //free(transition);
    [super dealloc];
}

- (NSString *) description
{
    return @"<dfa.description>";
}

<@errorMethod()>

@end /* end DFA<dfa.decisionNumber> implementation */<\n>
#pragma mark Cyclic DFA implementation end DFA<dfa.decisionNumber>

```

```

<\n>
>>
/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber, stateNumber, edges, needErrorClause, semPredState) ::= <<
/* cyclicDFAState */
NSInteger LA<decisionNumber>_<stateNumber> = [input LA:1];<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
NSInteger index<decisionNumber>_<stateNumber> = input.index;
[input rewind];<\n>
<endif>
s = -1;
<edges; separator="\nelse ">
<if(semPredState)> <! return input cursor to state before we rewound !>
[input seek.index<decisionNumber>_<stateNumber>];<\n>
<endif>
if ( s >= 0 )
    return s;
break;
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
/* cyclicDFAEdge */
if (<labelExpr><if(predicates)> && (<predicates>)<endif>) { s = <targetStateNumber>;}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n> /* eotDFAEdge */
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "<left>&&<right>"

orPredicates(operands) ::= "<operands; separator=\"||\">"

notPredicate(pred) ::= "!(<evalPredicate(pred, {})>)"

evalPredicate(pred,description) ::= "<pred>"

/*

```

```

* evalSynPredicate(pred,description) ::= "<pred>()"
*
* synpreds are broken in cyclic DFA special states
* Damn! For now, work around with using the selectors directly, and by providing a trampoline evalSynPred
method in
* DFA
*/
/* evalSynPredicate(pred,description) ::= "[self evaluateSyntacticPredicate:<pred>Selector stream:input]" */
evalSynPredicate(pred,description) ::= "[self evaluateSyntacticPredicate:@selector(<pred>_fragment)]"
/* evalSynPredicate(pred,description) ::= "[recognizer <pred>]" */

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber>==<atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
* somewhere. Must ask for the lookahead directly.
*/
isolatedLookaheadTest(atom,k,atomAsInt) ::= "[input LA:<k>] == <atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
(LA<decisionNumber>_<stateNumber> >= <lower> && LA<decisionNumber>_<stateNumber> \<= <upper>)
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(([input LA:<k>] >=
<lower>) && ([input LA:<k>] \<= <upper>))"

setTest(ranges) ::= <%
<ranges; separator="||">
%>

// A T T R I B U T E S

memVars(scope) ::= <% <scope.attributes:{a|<a.type> <a.name>;<\n>}; separator="\n"> %>

properties(scope) ::= <%
<scope.attributes:{a|@property (assign, getter=get<a.name>, setter=set<a.name>:) <a.type> <a.name>;<\n>};
separator="\n">
%>

methodsDecl(scope) ::= <%
<scope.attributes:{a|- (<a.type>)get<a.name>;<\n>- (void)set<a.name>:(<a.type>)aVal;<\n>}; separator="\n">
%>

synthesize(scope) ::= <% <scope.attributes:{a|@synthesize <a.name>;}; separator="\n"> %>

methods(scope) ::= <%
<scope.attributes:{a|
- (<a.type>)get<a.name> { return( <a.name> ); }\<\n>
- (void)set<a.name>:(<a.type>)aVal { <a.name> = aVal; }\<\n>}; separator="\n">

```

```

%>

globalAttributeScopeInterface(scope) ::= <%
/* globalAttributeScopeInterface */<\n>
@interface <scope.name>_Scope : SymbolsScope {<\n>
<if(scope.attributes)>
<memVars(scope)>
<endif>
}<\n>
<if(scope.attributes)>
/* start of globalAttributeScopeInterface properties */<\n>
<properties(scope)>
/* end globalAttributeScopeInterface properties */<\n>
<endif>

+ (<scope.name>_Scope *)new<scope.name>_Scope;<\n>
- (id) init;<\n>
<if(scope.attributes)>
/* start of globalAttributeScopeInterface methodsDecl */<\n>
<methodsDecl(scope)>
/* End of globalAttributeScopeInterface methodsDecl */<\n>
<endif>
@end /* end of <scope.name>_Scope interface */<\n>
%>

globalAttributeScopeMemVar(scope) ::= <%
/* globalAttributeScopeMemVar */<\n>
SymbolStack *<scope.name>_stack;<\n>
<scope.name>_Scope *<scope.name>_scope;<\n>
%>

globalAttributeScopeImplementation(scope) ::= <%
@implementation <scope.name>_Scope /* globalAttributeScopeImplementation */<\n>
<if(scope.attributes)>
/* start of synthesize -- OBJC-Line 1750 */<\n>
<synthesize(scope)><\n>
<endif>
<\n>
+ (<scope.name>_Scope *)new<scope.name>_Scope<\n>
{<\n>
return [[<scope.name>_Scope alloc] init];<\n>
}<\n>
<\n>
- (id) init<\n>
{<\n>
self = [super init];<\n>
return self;<\n>
}<\n>

```

```

<\n>
<if(scope.attributes)>
/* start of iterate get and set functions */<\n>
<methods(scope)><\n>
/* End of iterate get and set functions */<\n>
<endif>
@end /* end of <scope.name>_Scope implementation */<\n><\n>
%>

globalAttributeScopeInit(scope) ::= <<
/* globalAttributeScopeInit */<\n>
<scope.name>_scope = [<scope.name>_Scope new<scope.name>_Scope];<\n>
<scope.name>_stack = [SymbolStack newSymbolStackWithLen:30];<\n>
>>

globalAttributeScopeDealloc(scope) ::= <% [<scope.name>_stack release];<\n> %>

globalAttributeScope(scope) ::= <%
<if(scope.name)>
static <scope.name>_stack;<\n>
<endif>
%>

ruleAttributeScopeMemVar(scope) ::= <%
/* ObjC ruleAttributeScopeMemVar */<\n>
<if(scope.attributes)>
<scope.name>_Scope *<scope.name>_scope; /* ObjC ruleAttributeScopeMemVar */<\n>
<endif>
%>

ruleAttributeScopeInterface(scope) ::= <%
<if(scope.attributes)>
/* start of ruleAttributeScopeInterface */<\n>
@interface <scope.name>_Scope : SymbolsScope {<\n>
    <memVars(scope)><\n>
}<\n>
<\n>
/* start property declarations */<\n>
<properties(scope)><\n>
/* start method declarations */<\n>
+ (<scope.name>_Scope *)new<scope.name>_Scope;<\n>
- (id) init;<\n>
<methodsDecl(scope)><\n>
@end /* end of ruleAttributeScopeInterface */<\n><\n>
<endif>
%>

ruleAttributeScopeImplementation(scope) ::= <%

```

```

<if(scope.attributes)>
  @implementation <scope.name>_Scope /* start of ruleAttributeScopeImplementation */<\n>
  <synthesize(scope)><\n>
  <\n>
  + (<scope.name>_Scope *)new<scope.name>_Scope<\n>
  {<\n>
    return [[<scope.name>_Scope alloc] init];<\n>
  }<\n>
  <\n>
  - (id) init<\n>
  {<\n>
    self = [super init];<\n>
    return self;<\n>
  }<\n>
  <\n>
  /* start of <scope.name>_Scope get and set functions */<\n>
  <methods(scope)><\n>
  /* End of <scope.name>_Scope get and set functions */<\n>
  @end /* end of ruleAttributeScopeImplementation */<\n><\n>
<endif>
%>

```

```

ruleAttributeScopeInit(scope) ::= <%
/* ruleAttributeScopeInit */<\n>
<scope.name>_scope = [<scope.name>_Scope new<scope.name>_Scope];<\n>
<scope.name>_stack = [SymbolStack newSymbolStackWithLen:30];<\n>
%>

```

```

ruleAttributeScopeDealloc(scope) ::= <% [<scope.name>_Scope release];<\n> %>

```

```

ruleAttributeScope(scope) ::= <%
<if(scope.attributes)>
/* ruleAttributeScope */<\n>
static SymbolStack *<scope.name>_stack;<\n>
static <scope.name>_Scope *<scope.name>_scope;
<endif>
%>

```

```

ruleAttributeScopeDecl(scope) ::= <%
/* ruleAttributeScopeDecl */<\n>
<if(scope.attributes)>
<scope.name>_Scope *<scope.name>_scope;<\n>
<endif>
%>

```

```

returnStructName(r) ::= "<className()>_<r.name>_return"

```

```

returnType() ::= <%

```

```

<ifndef(!ruleDescriptor.isSynPred)>
<ifndef(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor:returnStructName()> *
<else>
<ifndef(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
<endif>
<else>
void
<endif>
%>

/** Generate the Objective-C type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <%
<ifndef(referencedRule.hasMultipleReturnValues)>
<className()>_<referencedRule.name>_return *<else>
<ifndef(referencedRule.hasSingleReturnValue)><referencedRule.singleValueReturnType><else>
void<endif>
<endif>
%>

delegateName(d) ::= <% <ifndef(d.label)><d.label><else>g<d.name><endif> %>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <% <objcTypeInitMap.(typeName)> %>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <%
<ruleLabelType(referencedRule=label.referencedRule)> <label.label.text> =
<initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;<\n>
%>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScopeInterface(scope) ::= <<
<ifndef(ruleDescriptor.hasMultipleReturnValues)>
/* returnScopeInterface <ruleDescriptor:returnStructName()> */
@interface <ruleDescriptor:returnStructName()> :
<ifndef(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope { /* returnScopeInterface line 1838 */

```



```

<@memVars()> /* ObjC start of memVars() */<\n>
<if(scope.attributes)>
<memVars(scope)><\n>
<endif>
}
/* start property declarations */
<@properties()><\n>
<if(scope.attributes)>
<properties(scope)><\n>
<endif>
/* start of method declarations */<\n>
+ (<ruleDescriptor:returnStructName()> *)new<ruleDescriptor:returnStructName()>;
/* this is start of set and get methods */
<@methodsDecl()> /* methodsDecl */<\n>
<if(scope.attributes)>
/* start of iterated get and set functions */<\n>
<methodsDecl(scope)><\n>
<endif>
@end /* end of returnScopeInterface interface */<\n>
<endif>
>>

returnScopeImplementation(scope) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
@implementation <ruleDescriptor:returnStructName()> /* returnScopeImplementation */<\n>
<@synthesize()> /* start of synthesize -- OBJC-Line 1837 */<\n>
<if(scope.attributes)>
<synthesize(scope)><\n>
<endif>
+ (<ruleDescriptor:returnStructName()> *)new<ruleDescriptor:returnStructName()><\n>
{<\n>
return [[(<ruleDescriptor:returnStructName()> alloc) init] retain];<\n>
}<\n>
<\n>
- (id) init<\n>
{<\n>
self = [super init];<\n>
return self;<\n>
}<\n>
<\n>
<@methods()><\n>
<if(scope.attributes)>
/* start of iterate get and set functions */<\n>
<methods(scope)><\n>
/* End of iterate get and set functions */<\n>
<endif>
<actions.(actionScope).ruleReturnMethods>
<@ruleReturnMembers()><\n>

```

```

@end /* end of returnScope implementation */<\n><\n>
<endif>
%>

parameterScope(scope) ::= <<
<! <scope.attributes:{ it | :(<it.type><it.name>); separator=" "> !>
<first(scope.attributes):{ a | :(<a.type><a.name>)> <rest(scope.attributes):{ a | arg<i>:(<a.type><a.name> }>
separator=" ">
>>

parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name> = <expr>";

/** Note that the scopeAttributeRef does not have access to the
 * grammar name directly
 */
scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
(((<scope>_Scope *) [<scope>_stack objectAtIndex:[<scope>_stack size]-<negIndex>-1])).<attr.name>
<else>
<if(index)>
((<scope>_Scope *) [<scope>_stack objectAtIndex:<index>]).<attr.name>
<else>
((<scope>_Scope *) [<scope>_stack peek]).<attr.name>
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
/* scopeSetAttributeRef */
<if(negIndex)>
((<scope>_Scope *) [<scope>_stack objectAtIndex:[<scope>_stack size]-<negIndex>-1]).<attr.name> = <expr>;
<else>
<if(index)>
((<scope>_Scope *) [<scope>_stack objectAtIndex:<index>]).<attr.name> = <expr>;
<else>
((<scope>_Scope *) [<scope>_stack peek]).<attr.name> = <expr>;
<endif>
<endif>
%>

scopeAttributeRefStack() ::= <<
/* scopeAttributeRefStack */
<if(negIndex)>
((<scope>_Scope *) [<scope>_stack objectAtIndex:[<scope>_stack count]-<negIndex>-1]).<attr.name> = <expr>;
<else>
<if(index)>
((<scope>_Scope *) [<scope>_stack objectAtIndex:<index>]).<attr.name> = <expr>;

```

```

<else>
(((<scope>_Scope *)[<scope>_stack peek]).<attr.name> = <expr>;
<endif>
<endif>
>>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size()>0 && $function::name.equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
(<scope>!=nil?<scope>.<attr.name>:<initValue(attr.type)>)
<else>
<scope>
<endif>
>>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> /* added to returnAttributeRef */<\n>
<else>
<attr.name><\n>
<endif>
%>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> =<expr>; /* added to returnSetAttributeRef */<\n>
<else>
<attr.name> = <expr>;<\n>
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

/* not sure the next are the right approach; and they are evaluated early; */
/* they cannot see TREE_PARSER or PARSER attributes for example. :( */

tokenLabelPropertyRef_text(scope,attr) ::= "(<scope>!=nil?<scope>.text:nil)"

```

```

tokenLabelPropertyRef_type(scope,attr) ::= "<scope>!=nil?<scope>.type:0)"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>!=nil?<scope>.line:0)"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>!=nil?<scope>.charPositionInLine:0)"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>!=nil?<scope>.channel:0)"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>!=nil?[<scope> getTokenIndex]:0)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "<scope>!=nil?[<scope>.text integerValue]:0)"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>!=nil?((<labelType> *)<scope>.start):nil)"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>!=nil?((<labelType> *)<scope>.stopToken):nil)"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>!=nil?((<ASTLabelType> *)<scope>.tree):nil)"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
<scope>!=nil?[[input getTokenStream] toStringFromStart:[[input getTreeAdaptor] getTokenStartIndex:[<scope>
getStart]]
    ToEnd:[[input getTreeAdaptor] getTokenStopIndex:[<scope> getStart]]]:0)
<else>
<scope>!=nil?([input toStringFromToken:[<scope> getStart] ToToken:[<scope> getStop]]):0)
<endif>
%>
ruleLabelPropertyRef_st(scope,attr) ::= "<scope>!=nil?[<scope> st]:nil)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "<scope>!=nil?<scope>.type:0)"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "<scope>!=nil?<scope>.line:0)"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "<scope>!=nil?<scope>.charPositionInLine:-1)"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "<scope>!=nil?<scope>.channel:0)"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "<scope>!=nil?[<scope> getTokenIndex]:0)"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "<scope>!=nil?<scope>.text:nil)"
lexerRuleLabelPropertyRef_int(scope,attr) ::= "<scope>!=nil?[<scope>.text integerValue]:0)"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "((<labelType> *)retval.start)"
rulePropertyRef_stop(scope,attr) ::= "((<labelType> *)retval.stopToken)"
rulePropertyRef_tree(scope,attr) ::= "((<ASTLabelType> *)retval.tree)"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
[[input getTokenStream] toStringFromStart:[[input getTreeAdaptor]
getTokenStartIndex:retval.start.token.startIndex]
    ToEnd:[[input getTreeAdaptor] getTokenStopIndex:retval.start.token.stopIndex]]
<else>
[input toStringFromToken:retval.start ToToken:[input LT:-1]]
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

```

```

/* hideous: find a way to cut down on the number of templates to support read/write access */
/* TODO: also, which ones are valid to write to? ask Ter */
lexerRuleSetPropertyRef_text(scope,attr,expr) ::= "state.text = <expr>";
lexerRuleSetPropertyRef_type(scope,attr,expr) ::= "_type"
lexerRuleSetPropertyRef_line(scope,attr,expr) ::= "state.tokenStartLine"
lexerRuleSetPropertyRef_pos(scope,attr,expr) ::= "state.tokenStartCharPositionInLine"
lexerRuleSetPropertyRef_index(scope,attr,expr) ::= "-1" /* undefined token index in lexer */
lexerRuleSetPropertyRef_channel(scope,attr,expr) ::= "state.channel=<expr>";
lexerRuleSetPropertyRef_start(scope,attr,expr) ::= "state.tokenStartCharIndex"
lexerRuleSetPropertyRef_stop(scope,attr,expr) ::= "(input.index-1)"

lexerRulePropertyRef_text(scope,attr) ::= "self.text"
lexerRulePropertyRef_type(scope,attr) ::= "state.type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(input.index-1)"
lexerRulePropertyRef_int(scope,attr) ::= "[<scope>.text integerValue]"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.start =<expr>";
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>"; /* "<\n>#error StringTemplates are
unsupported<\n>" */

/** How to execute an action */
execAction(action) ::= <<
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
    <action>
}
<else>
<action>
<endif>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
static ANTLRBitSet *<name>;
static const unsigned long long <name>_data[] = { <words64:{it | <it>LL};separator=", ">><\n>

```

>>

```
bitsetInit(name, words64) ::= <<  
<name> = [[ANTLRBitSet newBitSetWithBits:(const unsigned long long *)<name>_data  
Count:(NSUInteger)<length(words64)>] retain];<n>  
>>
```

```
codeFileExtension() ::= ".m"
```

```
true_value() ::= "YES"
```

```
false_value() ::= "NO"
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/ObjC/ObjC.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
Copyright (c) 2007-2008 Ronald Blaschke
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
group Perl5;
```

```

/** The overall file structure of a recognizer; stores methods for rules
 * and cyclic DFAs plus support code.
 */
outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,
           docComment, recognizer,
           name, tokens, tokenNames, rules, cyclicDFAs,
           bitsets, buildTemplate, buildAST, rewriteMode, profile,
           backtracking, synpreds, memoize, numRules,
           fileName, ANTLRVersion, generatedTimestamp, trace,
           scopes, superClass, literals) ::=
<<
# $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
<actions.(actionScope).header>

<@imports>
<if(TREE_PARSER)>
<endif>
<if(backtracking)>
<endif>
<@end>

<docComment>
<recognizer>
>>

lexer(grammar, name, tokens, scopes, rules, numRules, labelType="Token",
       filterMode, superClass="ANTLR::Runtime::Lexer") ::= <<
package <name>;

use Carp;
use English qw( -no_match_vars );
use Readonly;
use Switch;

use ANTLR::Runtime::BaseRecognizer;
use ANTLR::Runtime::DFA;
use ANTLR::Runtime::NoViableAltException;

use Moose;

extends 'ANTLR::Runtime::Lexer';

Readonly my $HIDDEN => ANTLR::Runtime::BaseRecognizer->HIDDEN;
sub HIDDEN { $HIDDEN }

use constant {
  <tokens:{ <it.name> => <it.type>, }; separator="\n">
};

```

```

<scopes:{<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
<actions.lexer.members>

sub BUILD {
    my ($self, $arg_ref) = @_ ;

    $self->init_dfas();
}

sub get_grammar_file_name {
    return "<fileName>";
}

<if(filterMode)>
<filteringNextToken()>
<endif>
<rules; separator="\n\n">

<synpreds:{ p | <lexerSynpred(p)>}>

<cyclicDFAs:{ dfa | has 'dfa<dfa.decisionNumber>';}; separator="\n">

sub init_dfas {
    my ($self) = @_ ;

    <cyclicDFAs:{ dfa |
        $self->dfa<dfa.decisionNumber>(<name::DFA<dfa.decisionNumber>->new({ recognizer => $self }));
    }; separator="\n">

    return;
}

<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

no Moose;
__PACKAGE__->meta->make_immutable();
1;

>>

perlTypeInitMap ::= [
    "$":"undef",
    "@":"()",
    "%":"()",
    default:"undef"
]

/** A override of Lexer.nextToken() that backtracks over mTokens() looking

```


- * for matches. No error can be generated upon error; just rewind, consume
- * a token and then try again. backtracking needs to be set as well.
- * Make rule memoization happen only at levels above 1 as we start mTokens
- * at backtracking==1.

*/

filteringNextToken() ::= <<

```

public Token nextToken() {
    while (true) {
        if ( input.LA(1)==CharStream.EOF ) {
            return Token.EOF_TOKEN;
        }
        token = null;
channel = Token.DEFAULT_CHANNEL;
        tokenStartCharIndex = input.index();
        tokenStartCharPositionInLine = input.getCharPositionInLine();
        tokenStartLine = input.getLine();
text = null;
        try {
            int m = input.mark();
            backtracking=1; <!-- means we won't throw slow exception !>
            failed=false;
            mTokens();
            backtracking=0;
            <!-- mTokens backtracks with synpred at backtracking==2
            and we set the synpredgate to allow actions at level 1. !>
            if ( failed ) {
                input.rewind(m);
                input.consume(); <!-- advance one char and try again !>
            }
            else {
                emit();
                return token;
            }
        }
        catch (RecognitionException re) {
            // shouldn't happen in backtracking mode, but...
            reportError(re);
            recover(re);
        }
    }
}

```

```

public void memoize(IntStream input,
    int ruleIndex,
    int ruleStartIndex)
{
    if ( backtracking>1 ) super.memoize(input, ruleIndex, ruleStartIndex);
}

```

```

public boolean alreadyParsedRule(IntStream input, int ruleIndex) {
if ( backtracking>1 ) return super.alreadyParsedRule(input, ruleIndex);
return false;
}
>>

actionGate() ::= "$self->state->backtracking==0"

filteringActionGate() ::= "backtracking==1"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass, filterMode,
    ASTLabelType="Object", labelType, members) ::= <<
package <name>;

use English qw( -no_match_vars );
use Readonly;
use Switch;
use Carp;
use ANTLR::Runtime::BitSet;

use Moose;

extends '<@superClassName><superClass><@end>';

Readonly my $token_names => [
    "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ">
];

use constant {
<tokens:{ <it.name> => <it.type>, }; separator="\n">
};

<bitsets:bitset(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},
    words64=it.bits)>

<scopes:{<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
<@members>
<! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>

sub BUILD {
    my ($self, $arg_ref) = @_;

<if(backtracking)>
    $self->state->rule_memo({});<\n>
<endif>

```

```

}
<@end>

sub get_token_names {
    return $token_names;
}

sub get_grammar_file_name {
    return "<fileName>";
}

<members>

<rules; separator="\n\n">

<synpreds:{ p | <synpred(p)>}>

<cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = __PACKAGE__::DFA<dfa.decisionNumber>->new($self);};
separator="\n">
<cyclicDFAs:cyclicDFA(> <! dump tables for all DFA !>

no Moose;
__PACKAGE__->meta->make_immutable();
1;
__END__
>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets, ASTLabelType,
superClass="ANTLR::Runtime::Parser", labelType="ANTLR::Runtime::Token",
members={<actions.parser.members>}) ::= <<
<genericParser(inputStreamType="ANTLR::Runtime::TokenStream", ...)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules, numRules, bitsets,
labelType={<ASTLabelType>}, ASTLabelType="Object", superClass="ANTLR::Runtime::TreeParser",
members={<actions.treeparser.members>}, filterMode) ::= <<
<genericParser(inputStreamType="TreeNodeStream", ...)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.

```

```

*/
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
# $ANTLR start <ruleName>
sub <ruleName>_fragment {
# <ruleDescriptor.parameterScope:parameterScope(scope=it)>

<if(trace)>
  $self->traceIn("<ruleName>_fragment", <ruleDescriptor.index>);
  eval {
    <block>
  };
  $self->traceOut("<ruleName>_fragment", <ruleDescriptor.index>);
  if ($EVAL_ERROR) {
    croak $EVAL_ERROR;
  }
<else>
  <block>
<endif>
}
# $ANTLR end <ruleName>
>>

```

```

synpred(name) ::= <<
public final boolean <name>() {
  backtracking++;
  <@start()>
  int start = input.mark();
  try {
    <name>_fragment(); // can never throw exception
  } catch (RecognitionException re) {
    System.err.println("impossible: "+re);
  }
  boolean success = !failed;
  input.rewind(start);
  <@stop()>
  backtracking--;
  failed=false;
  return success;
}<\n>
>>

```

```

lexerSynpred(name) ::= <<
<synpred(name)>
>>

```

```

ruleMemoization(name) ::= <<
<if(memoize)>

```

```

if ( backtracking>0 && alreadyParsedRule(input, <ruleDescriptor.index> ) ) { return <ruleReturnValue(); }
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>
if ($self->state->failed) {
    return <ruleReturnValue();
}
<endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (backtracking>0) { failed=true; return <ruleReturnValue();}<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(scope=ruleDescriptor.returnScope)>

# $ANTLR start <ruleName>
# <fileName>:<description>
sub <ruleName>() {
    my ($self, <ruleDescriptor.parameterScope:parameterScope(scope=it)>) = @_ ;
    <if(trace)>$self->traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
    <ruleLabelDefs()>
    <ruleDescriptor.actions.init>
    <@preamble()>
    eval {
        <ruleMemoization(name=ruleName)>
        <block>
        <ruleCleanUp()>
        <(ruleDescriptor.actions.after):execAction()>
    };
    <if(exceptions)>
        <exceptions:{e|<catch(decl=e.decl,action=e.action)><n}>>
    <else>
    <if(!emptyRule)>
    <if(actions.(actionScope).rulecatch)>
        <actions.(actionScope).rulecatch>
    <else>

```

```

my $exception = $EVAL_ERROR;
if (ref $exception && $exception->isa('ANTLR::Runtime::RecognitionException')) {
    $self->report_error($exception);
    $self->recover($self->input, $exception);
    $exception = undef;
}
<\n>
<endif>
<endif>
<endif>
<if(trace)>$self->traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
<memoize()>
<ruleScopeCleanUp()>
<finally>
if ($exception) {
    croak $exception;
    # $exception->rethrow();
}
<@postamble()>
return <ruleReturnValue()>;
}
# $ANTLR end <ruleName>
>>

catch(decl,action) ::= <<
catch (<e.decl>) {
    <e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
my $retval = <returnType()>->new();
$retval->set_start($self->input->LT(1));<\n>
<else>
<ruleDescriptor.returnScope.attributes:{ a |
my $<a.name> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
my $<ruleDescriptor.name>_start_index = $self->input->index();
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes:{<it>_stack.push(new <it>_scope());}; separator="\n">
<ruleDescriptor.ruleScope:{<it.name>_stack.push(new <it.name>_scope());}; separator="\n">
>>

```

```

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes:{<it>_stack.pop();}; separator="\n">
<ruleDescriptor.ruleScope:{<it.name>_stack.pop();}; separator="\n">
>>

```

```

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels]
: {my $<it.label.text> = undef;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]
: {List list_<it.label.text>=null;}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(label=it); separator="\n">
<ruleDescriptor.ruleListLabels: {ll|RuleReturnScope <ll.label.text> = null;}; separator="\n">
>>

```

```

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {<labelType> <it.label.text>=null;}; separator="\n"
>
<ruleDescriptor.charLabels: {my $<it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {List list_<it.label.text>=null;}; separator="\n"
>
>>

```

```

ruleReturnValue() ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
$<ruleDescriptor.singleValueReturnName>
<else>
$retval
<endif>
<endif>
<endif>
>>

```

```

ruleCleanUp() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
$retval->set_stop($self->input->LT(-1));<\n>
<endif>
<endif>

```

```

>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( backtracking>0 ) { memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }
<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
# $ANTLR start <ruleName>
sub m_<ruleName> {
# <ruleDescriptor.parameterScope:parameterScope(scope=it)>
my ($self) = @_ ;
<if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
<ruleDeclarations()>
eval {
<if(nakedBlock)>
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs()>
<ruleDescriptor.actions.init>
<block><\n>
<else>
my $_type = <ruleName>;
my $_channel = $self->DEFAULT_TOKEN_CHANNEL;
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs()>
<ruleDescriptor.actions.init>
<block>
<ruleCleanUp()>
$self->state->type($_type);
$self->state->channel($_channel);
<(ruleDescriptor.actions.after):execAction()>
<endif>
};
<if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
<memoize()>

if ($EVAL_ERROR) {
croak $EVAL_ERROR;
}
}
# $ANTLR end <ruleName>
>>

```



```

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
sub m_tokens {
  my ($self) = @_;
  <block><\n>
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
my $alt<decisionNumber> = <maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
<@prebranch()>
switch ($alt<decisionNumber>) {
  <alts:altSwitchCase()>
}
<@postbranch()>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
my $alt<decisionNumber> = <maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
switch ($alt<decisionNumber>) {
  <alts:altSwitchCase()>
}
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

```

```

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
my $cnt<decisionNumber> = 0;
<decls>
<@preloop()>
LOOP<decisionNumber>:
while (1) {
  my $alt<decisionNumber> = <maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  switch ($alt<decisionNumber>) {
    <alts:altSwitchCase()>
    else {
      if ( $cnt<decisionNumber> >= 1 ) { last LOOP<decisionNumber> }
      <ruleBacktrackFailure()>
      my $eee =
        ANTLR::Runtime::EarlyExitException->new(<decisionNumber>, $self->input);
      <@earlyExitException()>
      croak $eee;
    }
  }
  ++$cnt<decisionNumber>;
}
<@postloop()>
>>

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

```

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
# <fileName>:<description>
<decls>
<@preloop()>
LOOP<decisionNumber>:

```

```

while (1) {
  my $alt<decisionNumber> = <maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  switch ($alt<decisionNumber>) {
    <alts:altSwitchCase()>
    else { last LOOP<decisionNumber> }
  }
}
<@postloop()>
>>

```

```
closureBlockSingleAlt ::= closureBlock
```

```

/** Optional blocks (x)? are translated to (x|) by before code generation
 * so we can just use the normal block template
 */

```

```
optionalBlock ::= block
```

```
optionalBlockSingleAlt ::= block
```

```

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */

```

```

altSwitchCase() ::= <<
case <i> {
  <@prealt()>
  <it>
}<\n>
>>

```

```

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
# <fileName>:<description>
{
  <@declarations()>
  <elements:element()>
  <rew>
  <@cleanup()>
}
>>

```

```

/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

```

```

// ELEMENTS

/** Dump the elements one per line */
element() ::= <<
<@prematch(>
<it.el><\n>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>$<label> =<endif>$self->match($self->input, <token>,
$FOLLOW_<token>_in_<ruleName><elementIndex>);
<checkRuleBacktrackFailure(>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...>
<listLabel(elem=label,...>
>>

listLabel(label,elem) ::= <<
if (list_<label>==null) list_<label>=new ArrayList();
list_<label>.add(<elem>);<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = $self->input->LA(1);<\n>
<endif>
$self->match(<char>); <checkRuleBacktrackFailure(>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = $self->input->LA(1);<\n>
<endif>
$self->match_range(<a>,<b>); <checkRuleBacktrackFailure(>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label>= $self->input->LA(1);<\n>

```

```

<else>
<label>=(<labelType>)input.LT(1);<\n>
<endif>
<endif>
if ( <s> ) {
    $self->input->consume();
    <postmatchCode>
<if(!LEXER)>
    $self->state->error_recovery(0);
<endif>
    <if(backtracking)>failed=false;<endif>
}
else {
    <ruleBacktrackFailure()>
    my $mse =
        ANTLR::Runtime::MismatchedSetException->new(undef, $self->input);
    <@mismatchedSetException()>
<if(LEXER)>
    $self->recover($mse);
    $mse->throw();
<else>
    $mse->throw();
    <! use following code to make it recover inline; remove throw mse;
    $self->recoverFromMismatchedSet($self->input, $mse, $FOLLOW_set_in_<ruleName><elementIndex>);
    !>
<endif>
}<\n>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <<
<if(label)>
int <label>Start = getCharIndex();
$self->match(<string>); <checkRuleBacktrackFailure()>
<labelType> <label> = new CommonToken(input, Token.INVALID_TOKEN_TYPE,
Token.DEFAULT_CHANNEL, <label>Start, getCharIndex()-1);
<else>
$self->match(<string>); <checkRuleBacktrackFailure()><\n>
<endif>
>>

```

```
wildcard(label,elementIndex) ::= <<
<if(label)>
<label>=(<labelType>)input.LT(1);<\n>
<endif>
matchAny(input); <checkRuleBacktrackFailure()>
>>
```

```
wildcardAndListLabel(label,elementIndex) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>
```

```
/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = $self->input->LA(1);<\n>
<endif>
matchAny(); <checkRuleBacktrackFailure()>
>>
```

```
wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabel(elem=label,...)>
>>
```

```
/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
$self->push_follow($FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)>
$self-><rule.name>(<args; separator="," ">);<\n>
<else>
$self-><rule.name>(<args; separator="," ">);<\n>
<endif>
$self->state->_fsp($self->state->_fsp - 1);
<checkRuleBacktrackFailure()>
>>
```

```
/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>
```

```
/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
```

```

* is type Rule, whose toString is same: the rule name.
* Now though you can access full rule descriptor stuff.
*/
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
int <label>Start<elementIndex> = getCharIndex();
$self->m_<rule><(args; separator=" "> <checkRuleBacktrackFailure()>
<label> = new CommonToken(input, Token.INVALID_TOKEN_TYPE, Token.DEFAULT_CHANNEL,
<label>Start<elementIndex>, getCharIndex()-1);
<else>
$self->m_<rule.name><(args; separator=" "> <checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
int <label>Start<elementIndex> = getCharIndex();
match(EOF); <checkRuleBacktrackFailure()>
<labelType> <label> = new CommonToken(input, EOF, Token.DEFAULT_CHANNEL,
<label>Start<elementIndex>, getCharIndex()-1);
<else>
match(EOF); <checkRuleBacktrackFailure()>
<endif>
>>

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==Token.DOWN ) {
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
match(input, Token.UP, null); <checkRuleBacktrackFailure()>
}
<else>
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
match(input, Token.UP, null); <checkRuleBacktrackFailure()>
<endif>

```

```

>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(...)> ) ) {
    <ruleBacktrackFailure()>
    throw new FailedPredicateException(input, "<ruleName>", "<description>");
}
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
my $LA<decisionNumber>_<stateNumber> = $self->input->LA(<k>);<\n>
<edges; separator="\nls">
else {
<if(eotPredictsAlt)>
    $alt<decisionNumber> = <eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>
    my $nvae =
        ANTLR::Runtime::NoViableAltException->new({
            grammar_decision_description => "<description>",
            decision_number => <decisionNumber>,
            state_number => <stateNumber>,
            input => $self->input,
        });<\n>
    <@noViableAltException()>
    croak $nvae;<\n>
<endif>
}
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
my $LA<decisionNumber>_<stateNumber> = $self->input->LA(<k>);<\n>
<edges; separator="\nls">
>>

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).

```



```

* This is used only in the lexer so that for ('a')* on the end of a rule
* anything other than 'a' predicts exiting.
*/
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
my $LA<decisionNumber>_<stateNumber> = $self->input->LA(<k>);<\n>
<edges; separator="\nls"><\n>
<if(eotPredictsAlt)>
<if(!edges)>
$alt<decisionNumber> = <eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else {
    $alt<decisionNumber> = <eotPredictsAlt>;
}<\n>
<endif>
<endif>
>>

```

```

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "$alt<decisionNumber> = <alt>";

```

```

/** A simple edge with an expression. If the expression is satisfied,
* enter to the target state. To handle gated productions, we may
* have to evaluate some predicates for this edge.
*/
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) {
    <targetState>
}
>>

```

```
// F i x e d D F A (switch case)
```

```

/** A DFA state where a SWITCH may be generated. The code generator
* decides if this is possible: CodeGenerator.canGenerateSwitch().
*/
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( $self->input->LA(<k> ) ) {
    <edges; separator="\n">
    else {
    <if(eotPredictsAlt)>
        $alt<decisionNumber> = <eotPredictsAlt>;
    <else>
    <ruleBacktrackFailure()>
    my $nvae =
        ANTLR::Runtime::NoViableAltException->new({
            grammar_decision_description => "<description>",
            decision_number => <decisionNumber>,
            state_number => <stateNumber>,

```

```

        input => $self->input,
    });<\n>
    <@noViableAltException()>
    croak $nvae;<\n>
<endif>
}
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( $self->input->LA(<k> ) ) {
    <edges; separator="\n">
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( $self->input->LA(<k> ) ) {
    <edges; separator="\n"><\n>
    <if(eotPredictsAlt)>
    else { $alt<decisionNumber> = <eotPredictsAlt> }<\n>
    <endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
case [<labels: { <it> }; separator=", ">] { <targetState> }
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
$alt<decisionNumber> = $self->dfa<decisionNumber>->predict($self->input);
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
Readonly my $DFA<dfa.decisionNumber>_eot => ANTLR::Runtime::DFA->unpack_rle([

```

```

<dfa.javaCompressedEOT; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_eof => ANTLR::Runtime::DFA->unpack_rle([
<dfa.javaCompressedEOF; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_min => ANTLR::Runtime::DFA->unpack_rle([
<dfa.javaCompressedMin; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_max => ANTLR::Runtime::DFA->unpack_rle([
<dfa.javaCompressedMax; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_accept => ANTLR::Runtime::DFA->unpack_rle([
<dfa.javaCompressedAccept; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_special => ANTLR::Runtime::DFA->unpack_rle([
<dfa.javaCompressedSpecial; separator=", "> ]);
Readonly my $DFA<dfa.decisionNumber>_transition => [
<dfa.javaCompressedTransition:{s|ANTLR::Runtime::DFA->unpack_rle([ <s; separator=", "> ])}; separator=", "> ];

{
package <name>::DFA<dfa.decisionNumber>;
use ANTLR::Runtime::Class;

use strict;
use warnings;

extends 'ANTLR::Runtime::DFA';

sub BUILD {
    my $self = shift;
    my $param_ref = __PACKAGE__->unpack_params(@_, {
        spec => [
            {
                name => 'recognizer',
                isa => 'ANTLR::Runtime::BaseRecognizer'
            },
        ]
    });

    $self->recognizer($param_ref->{recognizer});
    $self->decision_number(<dfa.decisionNumber>);
    $self->eot($DFA<dfa.decisionNumber>_eot);
    $self->eof($DFA<dfa.decisionNumber>_eof);
    $self->min($DFA<dfa.decisionNumber>_min);
    $self->max($DFA<dfa.decisionNumber>_max);
    $self->accept($DFA<dfa.decisionNumber>_accept);
    $self->special($DFA<dfa.decisionNumber>_special);
    $self->transition($DFA<dfa.decisionNumber>_transition);
}

sub get_description {
    return "<dfa.description>";
}

```

```

<@errorMethod(>

<if(dfa.specialStateSTs)>
sub special_state_transition {
  my ($self, $param_ref) = unpack_params(@_, {
    spec => [
      {
        name => 's',
        type => SCALAR,
      },
      {
        name => 'input',
        isa => 'ANTLR::Runtime::IntStream',
      }
    ]
  });
  my $s = $param_ref->{s};
  my $input = $param_ref->{input};

  switch ($s) {
    <dfa.specialStateSTs:{state |
    case <i0> \{ <! compressed special state numbers 0..n-1 !>
      <state>}; separator="\n">
    }
  }

<if(backtracking)>
  if ($self->state->backtracking > 0) {
    $self->state->failed = 1;
    return -1;
  }<\n>
<endif>

  my $nvae =
    ANTLR::Runtime::NoViableAltException->new({
      grammar_decision_description => $self->get_description(),
      decision_number => <dfa.decisionNumber>,
      state_number => $s,
      input => $input,
    });<\n>
    $self->error($nvae);
    $nvae->throw();
  }<\n>
<endif>
}<\n>
>>

```

```

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
my $input = $self->input;
my $LA<decisionNumber>_<stateNumber> = $input->LA(1);<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
my $index<decisionNumber>_<stateNumber> = $input->index();
$input->rewind();<\n>
<endif>
s = -1;
<edges; separator="\nls">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>);<\n>
<endif>
if ( s>=0 ) return s;
break;
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) { s = <targetStateNumber>;}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "<left> && <right>"

orPredicates(operands) ::= "<operands; separator=\\|\\>"

notPredicate(pred) ::= "!(<evalPredicate(...)>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "$LA<decisionNumber>_<stateNumber> eq <atom>"

```

```

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "$self->input->LA(<k>) eq <atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <<
($LA<decisionNumber>_<stateNumber> ge <lower> && $LA<decisionNumber>_<stateNumber> le <upper>)
>>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "($self->input->LA(<k>) ge
<lower> && $self->input->LA(<k>) le <upper>)"

setTest(ranges) ::= "<ranges; separator=\\\" || \\\">"

// A T T R I B U T E S

globalAttributeScope(scope) ::= <<
<if(scope.attributes)>
protected static class <scope.name>_scope {
    <scope.attributes:{<it.decl>;}; separator=\\\"n" >
}
protected Stack <scope.name>_stack = new Stack();<n>
<endif>
>>

ruleAttributeScope(scope) ::= <<
<if(scope.attributes)>
protected static class <scope.name>_scope {
    <scope.attributes:{<it.decl>;}; separator=\\\"n" >
}
protected Stack <scope.name>_stack = new Stack();<n>
<endif>
>>

returnType() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.name>_return
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
<endif>
>>

/** Generate the Java type associated with a single or multiple return
 * values.

```

```

*/
ruleLabelType(referencedRule) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
<referencedRule.name>_return
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
<endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "undef".
 */
initValue(typeName) ::= <<
<if(typeName)>
<perlTypeInitMap.(typeName)>
<else>
undef
<endif>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
my $<label.label.text> = <initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;<\n>
>>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
{
package <returnType()>;
use ANTLR::Runtime::Class;

extends 'ANTLR::Runtime::<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope';

<scope.attributes:{public <it.decl>;}; separator="\n">
<@ruleReturnMembers()>
}
<endif>
>>

parameterScope(scope) ::= <<

```

```

<scope.attributes:{$<it.name>}; separator=", ">
>>

parameterAttributeRef(attr) ::= "$<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "$<attr.name> =<expr>";

scopeAttributeRef(scope,attr,index,negIndex) ::= <<
<if(negIndex)>
((<scope>_scope)<scope>_stack.elementAt(<scope>_stack.size()-<negIndex>-1)).<attr.name>
<else>
<if(index)>
((<scope>_scope)<scope>_stack.elementAt(<index>)).<attr.name>
<else>
((<scope>_scope)<scope>_stack.peek()).<attr.name>
<endif>
<endif>
>>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <<
<if(negIndex)>
((<scope>_scope)<scope>_stack.elementAt(<scope>_stack.size()-<negIndex>-1)).<attr.name> =<expr>;
<else>
<if(index)>
((<scope>_scope)<scope>_stack.elementAt(<index>)).<attr.name> =<expr>;
<else>
((<scope>_scope)<scope>_stack.peek()).<attr.name> =<expr>;
<endif>
<endif>
>>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size()>0 && $function::name.equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
$<scope>.<attr.name>
<else>
$<scope>
<endif>
>>

returnAttributeRef(ruleDescriptor,attr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>

```



```

<else>
$<attr.name>
<endif>
>>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> =<expr>;
<else>
$<attr.name> =<expr>;
<endif>
>>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "$<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "$<scope>->get_text()"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>.getType()"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>.getLine()"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>.getCharPositionInLine()"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>.getChannel()"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>.getTokenIndex()"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"

ruleLabelPropertyRef_start(scope,attr) ::= "(<labelType><scope>.start)"
ruleLabelPropertyRef_stop(scope,attr) ::= "(<labelType><scope>.stop)"
ruleLabelPropertyRef_tree(scope,attr) ::= "(<ASTLabelType><scope>.tree)"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
input.getTokenStream().toString(
input.getTreeAdaptor().getTokenStartIndex(<scope>.start),
input.getTreeAdaptor().getTokenStopIndex(<scope>.start))
<else>
substr($self->input, $<scope>->start, $<scope>->stop)
<endif>
>>

ruleLabelPropertyRef_st(scope,attr) ::= "<scope>.st"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "$<label>"

```

```

lexerRuleLabelPropertyRef_type(scope,attr) ::= "<scope>.getType()"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "<scope>.getLine()"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "<scope>.getCharPositionInLine()"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "<scope>.getChannel()"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "<scope>.getTokenIndex()"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "<scope>.getText()"

```

```

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "((<labelType>)retval.start)"
rulePropertyRef_stop(scope,attr) ::= "((<labelType>)retval.stop)"
rulePropertyRef_tree(scope,attr) ::= "((<ASTLabelType>)retval.tree)"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
input.getTokenStream().toString(
input.getTreeAdaptor().getTokenStartIndex(retval.start),
input.getTreeAdaptor().getTokenStopIndex(retval.start))
<else>
input.toString(retval.start,input.LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

```

```

lexerRulePropertyRef_text(scope,attr) ::= "getText()"
lexerRulePropertyRef_type(scope,attr) ::= "$_type"
lexerRulePropertyRef_line(scope,attr) ::= "tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "$_channel"
lexerRulePropertyRef_start(scope,attr) ::= "tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(getCharIndex()-1)"
lexerRulePropertyRef_self(scope,attr) ::= "$self"

```

```

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>";
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>";

```

```

/** How to execute an action */
execAction(action) ::= <<
<if(backtracking)>
<if(actions.(actionScope).synpredgate)>
if ( <actions.(actionScope).synpredgate> ) {
<action>
}
<else>
if ( backtracking==0 ) {
<action>

```

```

}
<endif>
<else>
<action>
<endif>
>>

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
Readonly my $<name> => ANTLR::Runtime::BitSet->new({ words64 => [ <words64:{'<it>'};separator=", "> ]
});<\n>
>>

```

```
codeFileExtension() ::= ".pm"
```

```
true() ::= "1"
false() ::= "0"
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Perl5/Perl5.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

/** Add a variable to track last element matched */

ruleDeclarations() ::= <<

<super.ruleDeclarations()>

<if(!ruleDescriptor.isSynPred)>

<ASTLabelType> _first_0 = null;

<ASTLabelType> _last = null;<\n>

<endif>

>>

/** What to emit when there is no rewrite rule. For auto build

* mode, does nothing.

*/

noRewrite(rewriteBlockLevel=false, treeLevel=false) ::= <<

<if(!ruleDescriptor.isSynPred)>

<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>

<if(rewriteMode)>

retval.tree = _first_0;

if (adaptor.getParent(retval.tree)!=null && adaptor.isNil(adaptor.getParent(retval.tree)))

retval.tree = (<ASTLabelType>)adaptor.getParent(retval.tree);

<endif>

<if(backtracking)>}<endif>

<endif>

>>

/** match ^(root children) in tree parser; override here to

* add tree construction actions.

*/

tree(root, actionsAfterRoot, children, nullableChildList,

enclosingTreeLevel, treeLevel) ::= <<

<if(!ruleDescriptor.isSynPred)>

_last = (<ASTLabelType>)input.LT(1);

{

```

<ASTLabelType> _save_last_<treeLevel> = _last;
<ASTLabelType> _first_<treeLevel> = null;
<if(!rewriteMode)>
<ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.nil();
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
<if(root.el.rule)>
if ( _first_<enclosingTreeLevel>==null ) _first_<enclosingTreeLevel> =
(<ASTLabelType>)<root.el.label>.getTree();
<elseif(root.el.label)>
if ( _first_<enclosingTreeLevel>==null ) _first_<enclosingTreeLevel> = <root.el.label>;
<endif>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==Token.DOWN ) {
  match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
  <children:element()>
  match(input, Token.UP, null); <checkRuleBacktrackFailure()>
}
<else>
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
match(input, Token.UP, null); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}<\n>
<else>
<super.tree(...)>
<endif>
>>

```

```
// TOKEN AST STUFF
```

```

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions={ }) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<else>
<super.tokenRefBang(...)>

```

```

<endif>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
  _last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
<else>
<super.tokenRef(...)>
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
<tokenRef(...)>
<listLabel(elem=label,...)>
<else>
<super.tokenRefAndListLabel(...)>
<endif>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
  _last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>

```

```

root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
<else>
<super.tokenRefRuleRoot(...)>
<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
<else>
<super.tokenRefRuleRootAndListLabel(...)>
<endif>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.wildcard(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.dupTree(<label>);
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
<else>
<super.wildcard(...)>
<endif>
>>

// SET AST

matchSet(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>

```

```

<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
<else>
<super.matchSet(...)>
<endif>
>>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={ }) ::= <<
<if(!ruleDescriptor.isSynPred)>
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
<else>
<super.matchRuleBlockSet(...)>
<endif>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(...)>
<else>
<super.matchSetBang(...)>
<endif>
>>

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={ }) ::= <<
<if(!ruleDescriptor.isSynPred)>
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
<else>
<super.matchSetRuleRoot(...)>
<endif>

```


>>

// RULE REF AST

/** rule auto construct */

```
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
  _last = (<ASTLabelType>)input.LT(1);
  <super.ruleRef(...)>
  <if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>
  <if(!rewriteMode)>
    adaptor.addChild(root_<treeLevel>, <label>.getTree());
  <else> <! rewrite mode !>
    if ( _first_<treeLevel>==null ) _first_<treeLevel> = (<ASTLabelType>)<label>.getTree();
  <endif>
  <else>
    <super.ruleRef(...)>
  <endif>
>>
```

/** x+=rule auto construct */

```
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
  <ruleRef(...)>
  <listLabel(label, {<label>.getTree()})>
  <else>
    <super.ruleRefAndListLabel(...)>
  <endif>
>>
```

/** ^(rule ...) auto construct */

```
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
  _last = (<ASTLabelType>)input.LT(1);
  <super.ruleRef(...)>
  <if(!rewriteMode)>
    <if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>root_<treeLevel> =
    (<ASTLabelType>)adaptor.becomeRoot(<label>.getTree(), root_<treeLevel>);
  <endif>
  <else>
    <super.ruleRefRuleRoot(...)>
  <endif>
>>
```

/** ^(x+=rule ...) auto construct */

```
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
  <ruleRefRuleRoot(...)>
```

```

<listLabel(label, {<label>.getTree()})>
<else>
<super.ruleRefRuleRootAndListLabel(...)>
<endif>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrackAndListLabel(...)>
<else>
<super.ruleRefTrackAndListLabel(...)>
<endif>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRootTrack(...)>
<else>
<super.ruleRefRootTrack(...)>
<endif>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRuleRootTrackAndListLabel(...)>
<else>
<super.ruleRefRuleRootTrackAndListLabel(...)>
<endif>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,args,terminalOptions={}) ::= <<
<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.nextNode())

```

```

<else>
stream_<token>.nextNode()
<endif>
>>

ruleCleanup() ::= <<
<super.ruleCleanup()>
<if(!ruleDescriptor.isSynPred)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<\n><endif>
retval.tree = (<ASTLabelType>)adaptor.rulePostProcessing(root_0);
<if(backtracking)>}<endif>
<endif>
<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Java/ASTTreeParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2010 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

```
THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

```

asTypeInitMap ::= [
    "int":"0",
    "uint":"0",
    "Number":"0.0",
    "Boolean":"false",
    default:"null" // anything other than an atomic type
]

/** The overall file structure of a recognizer; stores methods for rules
 * and cyclic DFAs plus support code.
 */
outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,
    docComment, recognizer,
    name, tokens, tokenNames, rules, cyclicDFAs,
    bitsets, buildTemplate, buildAST, rewriteMode, profile,
    backtracking, synpreds, memoize, numRules,
    fileName, ANTLRVersion, generatedTimestamp, trace,
    scopes, superClass, literals) ::=
<<
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
package<if(actions.(actionScope).package)> <actions.(actionScope).package><endif> {
    <actions.(actionScope).header>
    <@imports>
import org.antlr.runtime.*;
<if(TREE_PARSER)>
    import org.antlr.runtime.tree.*;
<endif>
    <@end>

    <docComment>
    <recognizer>
}
>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode, labelType="Token",
    superClass="Lexer") ::= <<
public class <grammar.recognizerName> extends
<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else><@superClassName><superClass>
<<@end><endif> {
    <tokens:{it |public static const <it.name>:int=<it.type>;}; separator="\n">
    <scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>
    <actions.lexer.members>

    // delegates
    <grammar.delegates:
        {g|public var <g.delegateName()>:<g.recognizerName>;}; separator="\n">
    // delegators

```

```

<grammar.delegators:
    { g|public var <g.delegateName():<g.recognizerName>;}; separator="\n">
<last(grammar.delegators):{ g|public var gParent:<g.recognizerName>;}>

    public function <grammar.recognizerName>(<grammar.delegators:{ g|<g.delegateName():<g.recognizerName>,
}>input:CharStream = null, state:RecognizerSharedState = null) {
        super(input, state);
        <cyclicDFAs:cyclicDFACTOR(>
<if(memoize)>
<if(grammar.grammarIsRoot)>
        this.state.ruleMemo = new Array(<numRules>+1);<n><! index from 1..n !>
<endif>
<endif>
        <grammar.directDelegates:
            { g|<g.delegateName()> = new <g.recognizerName>(<trunc(g.delegators):{ p|<p.delegateName()>, }>this,
input, this.state);}; separator="\n">
        <grammar.delegators:
            { g|this.<g.delegateName()> = <g.delegateName()>;}; separator="\n">
            <last(grammar.delegators):{ g|gParent = <g.delegateName()>;}>
        }
        public override function get grammarFileName():String { return "<fileName>"; }

<if(filterMode)>
    <filteringNextToken()>
<endif>
    <rules; separator="\n\n">

    <synpreds:{ p | <lexerSynpred(p)>}>

    <cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

}
>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */
filteringNextToken() ::= <<
public override function nextToken():Token {
    while (true) {
        if ( input.LA(1)==CharStreamConstants.EOF ) {
            return TokenConstants.EOF_TOKEN;
        }
        this.state.token = null;
        this.state.channel = TokenConstants.DEFAULT_CHANNEL;
    }
}

```

```

this.state.tokenStartCharIndex = input.index;
this.state.tokenStartCharPositionInLine = input.charPositionInLine;
this.state.tokenStartLine = input.line;
this.state.text = null;
try {
    var m:int = input.mark();
    this.state.backtracking=1; <! means we won't throw slow exception !>
    this.state.failed=false;
    mTokens();
    this.state.backtracking=0;
    <! mTokens backtracks with synpred at backtracking==2
    and we set the synpredgate to allow actions at level 1. !>
    if ( this.state.failed ) {
        input.rewindTo(m);
        input.consume(); <! advance one char and try again !>
    }
    else {
        emit();
        return this.state.token;
    }
}
catch (re:RecognitionException) {
    // shouldn't happen in backtracking mode, but...
    reportError(re);
    recover(re);
}
}
// Not reached - For ActionScript compiler
throw new Error();
}

public override function memoize(input:IntStream,
    ruleIndex:int,
    ruleStartIndex:int):void
{
    if ( this.state.backtracking>1 ) super.memoize(input, ruleIndex, ruleStartIndex);
}

public override function alreadyParsedRule(input:IntStream, ruleIndex:int):Boolean {
    if ( this.state.backtracking>1 ) return super.alreadyParsedRule(input, ruleIndex);
    return false;
}
}
>>

actionGate() ::= "this.state.backtracking==0"

filteringActionGate() ::= "this.state.backtracking==1"

```

```

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass,
    labelType, members, rewriteElementType,
    filterMode, ASTLabelType="Object") ::= <<
public class <grammar.recognizerName> extends
<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else><@superClassName><superClass>
<<@end><endif> {
<if(grammar.grammarIsRoot)>
    public static const tokenNames:Array = [
        "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ">
    ];<\n>
<endif>
    <tokens:{it |public static const <it.name>:int=<it.type>;}; separator="\n">

    // delegates
    <grammar.delegates: {g|public var <g:delegateName():<g.recognizerName>;}; separator="\n">
    // delegators
    <grammar.delegators:
        {g|public var <g:delegateName():<g.recognizerName>;}; separator="\n">
    <last(grammar.delegators):{g|public var gParent:<g.recognizerName>;}>

    <scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>
    <@members>
    <! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
    public function <grammar.recognizerName>(<grammar.delegators:{g|<g:delegateName():<g.recognizerName>,
}>input:<inputStreamType>, state:RecognizerSharedState = null) {
        super(input, state);
        <cyclicDFAs:cyclicDFACtor>
        <parserCtorBody>
        <grammar.directDelegates:
            {g|<g:delegateName()> = new <g.recognizerName>(<trunc(g.delegators):{p|<p:delegateName()>, }>this,
input, this.state);}; separator="\n">
            <grammar.indirectDelegates:{g | <g:delegateName()> = <g.delegator:delegateName()>.<g:delegateName()>;};
separator="\n">
            <last(grammar.delegators):{g|gParent = <g:delegateName()>;}>
        }
    <@end>

    public override function get tokenNames():Array { return
<grammar.composite.rootGrammar.recognizerName>.tokenNames; }
    public override function get grammarFileName():String { return "<fileName>"; }

    <members>

    <rules; separator="\n\n">

    <! generate rule/method definitions for imported rules so they

```

```

    appear to be defined in this recognizer. !>
    // Delegated rules
    <grammar.delegatedRules:{ruleDescriptor|
        public function <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope():<returnType()> {
    <if(ruleDescriptor.hasReturnValue)>return
    <endif><ruleDescriptor.grammar:delegateName()>.<ruleDescriptor.name>(<ruleDescriptor.parameterScope.attributes:
    {a|<a.name>}; separator=", ">); \} }; separator="\n">

    <synpreds:{p | <synpred(p)>>

    <cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

    <bitsets:{it | <bitset(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},
        words64=it.bits)>>
    }
    >>

    parserCtorBody() ::= <<
    <if(memoize)>
    <if(grammar.grammarIsRoot)>
    this.state.ruleMemo = new Array(<length(grammar.allImportedRules)>+1);<\n> <! index from 1..n !>
    <endif>
    <endif>
    <grammar.delegators:
    {g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
    >>

    parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets, ASTLabelType="Object",
    superClass="Parser", labelType="Token", members={<actions.parser.members>}) ::= <<
    <genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
        bitsets, "TokenStream", superClass,
        labelType, members, "Token",
        false, ASTLabelType)>
    >>

    /** How to generate a tree parser; same as parser except the input
    * stream is a different type.
    */
    treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
        numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="Object",
        superClass="TreeParser", members={<actions.treeparser.members>}) ::= <<
    <genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
        bitsets, "TreeNodeStream", superClass,
        labelType, members, "Node",
        filterMode, ASTLabelType)>
    >>

    /** A simpler version of a rule template that is specific to the imaginary

```



```

* rules created for syntactic predicates. As they never have return values
* nor parameters etc..., just give simplest possible method. Don't do
* any of the normal memoization stuff in here either; it's a waste.
* As predicates cannot be inlined into the invoking rule, they need to
* be in a rule by themselves.
*/
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start <ruleName>
public final function <ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope()>):void {
    <ruleLabelDefs()>
<if(trace)>
    traceIn("<ruleName>_fragment", <ruleDescriptor.index>);
    try {
        <block>
    }
    finally {
        traceOut("<ruleName>_fragment", <ruleDescriptor.index>);
    }
<else>
    <block>
<endif>
}
// $ANTLR end <ruleName>
>>

synpred(name) ::= <<
public final function <name>():Boolean {
    this.state.backtracking++;
    <@start()>
    var start:int = input.mark();
    try {
        <name>_fragment(); // can never throw exception
    } catch (re:RecognitionException) {
        trace("impossible: "+re);
    }
    var success:Boolean = !this.state.failed;
    input.rewindTo(start);
    <@stop()>
    this.state.backtracking--;
    this.state.failed=false;
    return success;
}<\n>
>>

lexerSynpred(name) ::= <<
<synpred(name)>
>>

```

```

ruleMemoization(name) ::= <<
<if(memoize)>
if ( this.state.backtracking>0 && alreadyParsedRule(input, <ruleDescriptor.index>) ) { return <ruleReturnValue()>;
}
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>if (this.state.failed) return <ruleReturnValue()>;<endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (this.state.backtracking>0) {this.state.failed=true; return <ruleReturnValue()>;}<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
// $ANTLR start <ruleName>
// <fileName>:<description>
public final function <ruleName>(<ruleDescriptor.parameterScope:parameterScope()>):<returnType()> {
  <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
  <ruleScopeSetUp()>
  <ruleDeclarations()>
  <ruleLabelDefs()>
  <ruleDescriptor.actions.init>
  <@preamble()>
  try {
    <ruleMemoization(name=ruleName)>
    <block>
    <ruleCleanUp()>
    <(ruleDescriptor.actions.after):execAction()>
  }
  <if(exceptions)>
    <exceptions:{e|<catch(decl=e.decl,action=e.action)><n>}>
  <else>
    <if(!emptyRule)>
    <if(actions.(actionScope).rulecatch)>
      <actions.(actionScope).rulecatch>
    <else>
      catch (re:RecognitionException) {
        reportError(re);
        recoverStream(input,re);
      }
    </if>
  </if>
}
}

```

```

    <@setErrorReturnValue()>
    }<\n>
<endif>
<endif>
<endif>
    finally {
        <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
        <memoize()>
        <ruleScopeCleanUp()>
        <finally>
    }
    <@postamble()>
    return <ruleReturnValue()>;
}
// $ANTLR end <ruleName>
>>

catch(decl,action) ::= <<
catch (<e.decl>) {
    <e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
var retval:<returnType()> = new <returnType()>();
retval.start = input.LT(1);<\n>
<else>
<ruleDescriptor.returnScope.attributes: { a |
var <a.name>:<a.type> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
var <ruleDescriptor.name>_startIndex:int = input.index;
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: {it |<it>_stack.push(new Object());}; separator="\n">
<ruleDescriptor.ruleScope: {it |<it.name>_stack.push(new Object());}; separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: {it |<it>_stack.pop();}; separator="\n">
<ruleDescriptor.ruleScope: {it |<it.name>_stack.pop();}; separator="\n">
>>

ruleLabelDefs() ::= <<

```

```

<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it |var <it.label.text>:<labelType>=null;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: {it |var list_<it.label.text>:Array=null;}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels: {ll|var <ll.label.text>:RuleReturnScope = null;}; separator="\n">
>>

```

```

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it |var <it.label.text>:<labelType>=null;}; separator="\n"
>
<ruleDescriptor.charLabels: {it |var <it.label.text>:int;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]
: {it |var list_<it.label.text>:Array=null;}; separator="\n"
>
>>

```

```

ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
%>

```

```

ruleCleanup() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.stop = input.LT(-1);<\n>
<endif>
<endif>
>>

```

```

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( this.state.backtracking>0 ) { memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }

```

```

<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
// $ANTLR start <ruleName>
public final function m<ruleName>(<ruleDescriptor.parameterScope:parameterScope(>):void {
    <if(trace)>traceIn("<ruleName>", <ruleDescriptor.index>);<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
    try {
<if(nakedBlock)>
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block><\n>
    <else>
        var _type:int = <ruleName>;
        var _channel:int = DEFAULT_TOKEN_CHANNEL;
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block>
        <ruleCleanUp()>
        this.state.type = _type;
        this.state.channel = _channel;
        <(ruleDescriptor.actions.after):execAction()>
    <endif>
    }
    finally {
        <if(trace)>traceOut("<ruleName>", <ruleDescriptor.index>);<endif>
        <ruleScopeCleanUp()>
        <memoize()>
    }
}
// $ANTLR end <ruleName>
>>

```

```

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
public override function mTokens():void {
    <block><\n>
}

```

>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */

block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

var alt<decisionNumber>:int=<maxAlt>;

<decls>

<@predecision()>

<decision>

<@postdecision()>

<@prebranch()>

switch (alt<decisionNumber>) {

<alts:{a | <altSwitchCase(i, a)>}>

}

<@postbranch()>

>>

/** A rule block with multiple alternatives */

ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>

var alt<decisionNumber>:int=<maxAlt>;

<decls>

<@predecision()>

<decision>

<@postdecision()>

switch (alt<decisionNumber>) {

<alts:{a | <altSwitchCase(i, a)>}>

}

>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

// <fileName>:<description>

<decls>

<@prealt()>

<alts>

<@postalt()>

>>

/** A special case of a (...) subrule with a single alternative */

blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

// <fileName>:<description>

<decls>

<@prealt()>

<alts>

<@postalt()>

>>

```

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
var cnt<decisionNumber>:int=0;
<decls>
<@preloop()>
loop<decisionNumber>:
do {
  var alt<decisionNumber>:int=<maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  switch (alt<decisionNumber>) {
  <alts:{a | <altSwitchCase(i, a)}>
  default :
    if ( cnt<decisionNumber> >= 1 ) break loop<decisionNumber>;
    <ruleBacktrackFailure()>
    throw new EarlyExitException(<decisionNumber>, input);
    <! Need to add support for earlyExitException debug hook !>
  }
  cnt<decisionNumber>++;
} while (true);
<@postloop()>
>>

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

```

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
// <fileName>:<description>
<decls>
<@preloop()>
loop<decisionNumber>:
do {
  var alt<decisionNumber>:int=<maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  switch (alt<decisionNumber>) {
  <alts:{a | <altSwitchCase(i, a)}>
  default :
    break loop<decisionNumber>;
  }
} while (true);
<@postloop()>

```

```

>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation
 * so we can just use the normal block template
 */
optionalBlock ::= block

optionalBlockSingleAlt ::= block

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */
altSwitchCase(altNum, alt) ::= <<
case <altNum> :
  <@prealt()>
  <alt>
  break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
}
>>

/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(e) ::= <<
<@prematch()>
<e.el><\n>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<

```



```

<if(label)><label>=<labelType><endif>matchStream(input,<token>,FOLLOW_<token>_in_<ruleName><element
Index><if(label)><endif>; <checkRuleBacktrackFailure()>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(token,label,elementIndex,terminalOptions)>
<listLabel(label, label)>
>>

listLabel(label,elem) ::= <<
if (list_<label>==null) list_<label>=new Array();
list_<label>.push(<elem>);<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
match(<char>); <checkRuleBacktrackFailure()>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
matchRange(<a>,<b>); <checkRuleBacktrackFailure()>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,terminalOptions,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label>= input.LA(1);<\n>
<else>
<label>=<labelType>(input.LT(1));<\n>
<endif>
<endif>
if ( <s> ) {
    input.consume();
    <postmatchCode>
<if(!LEXER)>
    this.state.errorRecovery=false;
<endif>
    <if(backtracking)>this.state.failed=false;<endif>
}

```

```

else {
    <ruleBacktrackFailure()>
    <@mismatchedSetException()>
<if(LEXER)>
    throw recover(new MismatchedSetException(null,input));<\n>
<else>
    throw new MismatchedSetException(null,input);
    <! use following code to make it recover inline; remove throw mse;
    recoverFromMismatchedSet(input,mse,FOLLOW_set_in_<ruleName><elementIndex>);
    !>
<endif>
}<\n>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(label, label)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex="0") ::= <<
<if(label)>
var <label>Start:int = charIndex;
matchString(<string>); <checkRuleBacktrackFailure()>
var <label>StartLine<elementIndex>:int = line;
var <label>StartCharPos<elementIndex>:int = charPositionInLine;
<label> = CommonToken.createFromStream(input, TokenConstants.INVALID_TOKEN_TYPE,
TokenConstants.DEFAULT_CHANNEL, <label>Start, charIndex-1);
<label>.line = <label>StartLine<elementIndex>;
<label>.charPositionInLine = <label>StartCharPos<elementIndex>;
<else>
matchString(<string>); <checkRuleBacktrackFailure()><\n>
<endif>
>>

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>
<label>=<labelType>(input.LT(1));<\n>
<endif>
matchAny(input); <checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<wildcard(...)>
<listLabel(label, label)>
>>

```

```

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
matchAny(); <checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(label, elementIndex)>
<listLabel(label, label)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 *
 * GMS: Note: do not use post-decrement operator! ASC produces bad code for exceptions in this case.
 * See: https://bugs.adobe.com/jira/browse/ASC-3625
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
pushFollow(FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)><scope.delegateName()>.<endif><rule.name>(<args; separator=", ">);<\n>
state._fsp = state._fsp - 1;
<checkRuleBacktrackFailure()>
>>

/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, label)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
var <label>Start<elementIndex>:int = charIndex;
var <label>StartLine<elementIndex>:int = line;
var <label>StartCharPos<elementIndex>:int = charPositionInLine;

```

```

<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<label> = CommonToken.createFromStream(input, TokenConstants.INVALID_TOKEN_TYPE,
TokenConstants.DEFAULT_CHANNEL, <label>Start<elementIndex>, charIndex-1);
<label>.line = <label>StartLine<elementIndex>;
<label>.charPositionInLine = <label>StartCharPos<elementIndex>;
<else>
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<endif>
>>

```

```

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(rule,label,args,elementIndex,scope)>
<listLabel(label, label)>
>>

```

```

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
var <label>Start<elementIndex>:int = charIndex;
var <label>StartLine<elementIndex>:int = line;
var <label>StartCharPos<elementIndex>:int = charPositionInLine;
match(EOF); <checkRuleBacktrackFailure()>
var <label>:<labelType> = CommonToken.createFromStream(input, EOF,
TokenConstants.DEFAULT_CHANNEL, <label>Start<elementIndex>, charIndex-1);
<label>.line = <label>StartLine<elementIndex>;
<label>.charPositionInLine = <label>StartCharPos<elementIndex>;
<else>
match(EOF); <checkRuleBacktrackFailure()>
<endif>
>>

```

```

// used for left-recursive rules
recRuleDefArg()          ::= "var <recRuleArg()>:int"
recRuleArg()             ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> \<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name) ::= "$<name>=$<src>.<name>;"

```

```

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==TokenConstants.DOWN ) {

```

```

    matchStream(input, TokenConstants.DOWN, null); <checkRuleBacktrackFailure()>
    <children:element()>
    matchStream(input, TokenConstants.UP, null); <checkRuleBacktrackFailure()>
}
<else>
matchStream(input, TokenConstants.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
matchStream(input, TokenConstants.UP, null); <checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(pred,description)>) ) {
    <ruleBacktrackFailure()>
    throw new FailedPredicateException(input, "<ruleName>", "<description>");
}
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
var LA<decisionNumber>_<stateNumber>:int = input.LA(<k>);<n>
<edges; separator="\nelse ">
else {
<if(eotPredictsAlt)>
    alt<decisionNumber>=<eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>
    throw new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input);<n>
    <! Need to add hook for noViableAltException() !>
<endif>
}
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
var LA<decisionNumber>_<stateNumber>:int = input.LA(<k>);<n>
<edges; separator="\nelse ">
>>

/** A DFA state that is actually the loopback decision of a closure

```

```

* loop. If end-of-token (EOT) predicts any of the targets then it
* should act like a default clause (i.e., no error can be generated).
* This is used only in the lexer so that for ('a'* on the end of a rule
* anything other than 'a' predicts exiting.
*/
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
var LA<decisionNumber>_<stateNumber>:int = input.LA(<k>);<\n>
<edges; separator="\nelse "><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber>=<eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else {
    alt<decisionNumber>=<eotPredictsAlt>;
}<\n>
<endif>
<endif>
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>";

/** A simple edge with an expression. If the expression is satisfied,
* enter to the target state. To handle gated productions, we may
* have to evaluate some predicates for this edge.
*/
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) {
    <targetState>
}
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
* decides if this is possible: CodeGenerator.canGenerateSwitch().
*/
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k> ) ) {
<edges; separator="\n">
default:
<if(eotPredictsAlt)>
    alt<decisionNumber>=<eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>
    throw new NoViableAltException("<description>", <decisionNumber>, <stateNumber>, input);<\n>
    <! Need to add hook for noViableAltException !>
<endif>

```

```

}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k>) ) {
    <edges; separator="\n">
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( input.LA(<k>) ) {
<edges; separator="\n"><\n>
<if(eotPredictsAlt)>
default:
    alt<decisionNumber>=<eotPredictsAlt>;
    break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels: {it |case <it>:}; separator="\n">
{
    <targetState>
}
break;
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = dfa<decisionNumber>.predict(input);
>>

cyclicDFActor(dfa) ::= <<

dfa<dfa.decisionNumber> = new DFA(this, <dfa.decisionNumber>,
    "<dfa.description>",
    DFA<dfa.decisionNumber>_eot, DFA<dfa.decisionNumber>_eof, DFA<dfa.decisionNumber>_min,
    DFA<dfa.decisionNumber>_max, DFA<dfa.decisionNumber>_accept, DFA<dfa.decisionNumber>_special,
    DFA<dfa.decisionNumber>_transition<if(dfa.specialStateSTs)>,
    DFA<dfa.decisionNumber>_specialStateTransition<endif>);

>>

```

```

/* Dump DFA tables as run-length-encoded Strings of octal values.
* Can't use hex as compiler translates them before compilation.
* These strings are split into multiple, concatenated strings.
* Java puts them back together at compile time thankfully.
* Java cannot handle large static arrays, so we're stuck with this
* encode/decode approach. See analysis and runtime DFA for
* the encoding methods.
*/
cyclicDFA(dfa) ::= <<

private const DFA<dfa.decisionNumber>_eot:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedEOT; wrap=\""+\n  \>");
private const DFA<dfa.decisionNumber>_eof:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedEOF; wrap=\""+\n  \>");
private const DFA<dfa.decisionNumber>_min:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedMin; wrap=\""+\n  \>", true);
private const DFA<dfa.decisionNumber>_max:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedMax; wrap=\""+\n  \>", true);
private const DFA<dfa.decisionNumber>_accept:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedAccept; wrap=\""+\n  \>");
private const DFA<dfa.decisionNumber>_special:Array =
  DFA.unpackEncodedString("<dfa.javaCompressedSpecial; wrap=\""+\n  \>");
private const DFA<dfa.decisionNumber>_transition:Array = [
  <dfa.javaCompressedTransition: {s|DFA.unpackEncodedString("<s; wrap=\""+\n\>"); separator=",\n">
];
<if(dfa.specialStateSTs)>
private function DFA<dfa.decisionNumber>_specialStateTransition(dfa:DFA, s:int, _input:InputStream):int {
  <if(LEXER)>
  var input:InputStream = _input;
  <endif>
  <if(PARSER)>
  var input:TokenStream = TokenStream(_input);
  <endif>
  <if(TREE_PARSER)>
  var input:TreeNodeStream = TreeNodeStream(_input);
  <endif>
  var _s:int = s;
  switch ( s ) {
  <dfa.specialStateSTs: {state |
  case <i0> : <! compressed special state numbers 0..n-1 !>
    <state>}; separator="\n">
  }
<if(backtracking)>
  if (this.state.backtracking>0) {this.state.failed=true; return -1;}<\n>
<endif>
  throw dfa.error(new NoViableAltException(dfa.description, <dfa.decisionNumber>, _s, input));
}<\n>
<endif>

```



```

protected var dfa<dfa.decisionNumber>:DFA; // initialized in constructor

>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
var LA<decisionNumber>_<stateNumber>:int = input.LA(1);<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
var index<decisionNumber>_<stateNumber>:int = input.index;
input.rewind();<\n>
<endif>
s = -1;
<edges; separator="\nelse ">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>);<\n>
<endif>
if ( s>=0 ) return s;
break;
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> ) { s = <targetStateNumber>;}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "<left>&&<right>"

orPredicates(operands) ::= "<operands; separator=\\|\\>"

notPredicate(pred) ::= "!(<evalPredicate(pred,{}>)"

evalPredicate(pred,description) ::= "<pred>"

```

```

evalSynPredicate(pred,description) ::= "<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber>===<atomAsInt>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "input.LA(<k>)==<atomAsInt>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
(LA<decisionNumber>_<stateNumber> >= <lowerAsInt> && LA<decisionNumber>_<stateNumber> \<=
<upperAsInt>)
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(input.LA(<k>) >=
<lowerAsInt> && input.LA(<k>) \<= <upperAsInt>)"

setTest(ranges) ::= <<
<ranges; separator="|">
>>

// A T T R I B U T E S

globalAttributeScope(scope) ::= <<
<if(scope.attributes)>
protected var <scope.name>_stack:Array = new Array();<\n>
<endif>
>>

ruleAttributeScope(scope) ::= <<
<if(scope.attributes)>
protected var <scope.name>_stack:Array = new Array();<\n>
<endif>
>>

returnStructName() ::= "<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope"

returnType() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnStructName()>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
<endif>
>>

```

```

/** Generate the Java type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
<returnStructName()>
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
<endif>
>>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<asTypeInitMap.(typeName)>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
var <label.label.text>:<ruleLabelType(referencedRule=label.referencedRule)> =
<initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;<\n>
>>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
public static class <returnType()> extends <if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope {
  <scope.attributes:{it |public <it.decl>;}; separator="\n">
  <@ruleReturnMembers()>
};
<endif>
>>

parameterScope(scope) ::= <<
<scope.attributes:{it |<it.name>:<it.type>;}; separator=" ">

```

```
>>
```

```
parameterAttributeRef(attr) ::= "<attr.name>"  
parameterSetAttributeRef(attr,expr) ::= "<attr.name> =<expr>";
```

```
scopeAttributeRef(scope,attr,index,negIndex) ::= <<  
<if(negIndex)>  
<scope>_stack[<scope>_stack.length-<negIndex>-1].<attr.name>  
<else>  
<if(index)>  
<scope>_stack[<index>].<attr.name>  
<else>  
<scope>_stack[<scope>_stack.length-1].<attr.name>  
<endif>  
<endif>  
>>
```

```
scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <<  
<if(negIndex)>  
<scope>_stack[<scope>_stack.length-<negIndex>-1].<attr.name> =<expr>;  
<else>  
<if(index)>  
<scope>_stack[<index>].<attr.name> =<expr>;  
<else>  
<scope>_stack[<scope>_stack.length-1].<attr.name> =<expr>;  
<endif>  
<endif>  
>>
```

```
/** $x is either global scope or x is rule with dynamic scope; refers  
 * to stack itself not top of stack. This is useful for predicates  
 * like {$function.size()>0 && $function::name.equals("foo")}?  
 */  
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"
```

```
/** reference an attribute of rule; might only have single return value */  
ruleLabelRef(referencedRule,scope,attr) ::= <<  
<if(referencedRule.hasMultipleReturnValues)>  
(<scope>!=null?<scope>.values.<attr.name>:<initValue(attr.type)>)  
<else>  
<scope>  
<endif>  
>>
```

```
returnAttributeRef(ruleDescriptor,attr) ::= <<  
<if(ruleDescriptor.hasMultipleReturnValues)>  
retval.values.<attr.name>  
<else>
```

```

<attr.name>
<endif>
>>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.values.<attr.name> =<expr>;
<else>
<attr.name> =<expr>;
<endif>
>>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "(<scope>!=null?<scope>.text:null)"
tokenLabelPropertyRef_type(scope,attr) ::= "(<scope>!=null?<scope>.type:0)"
tokenLabelPropertyRef_line(scope,attr) ::= "(<scope>!=null?<scope>.line:0)"
tokenLabelPropertyRef_pos(scope,attr) ::= "(<scope>!=null?<scope>.charPositionInLine:0)"
tokenLabelPropertyRef_channel(scope,attr) ::= "(<scope>!=null?<scope>.channel:0)"
tokenLabelPropertyRef_index(scope,attr) ::= "(<scope>!=null?<scope>.tokenIndex:0)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "(<scope>!=null?int(<scope>.text):0)"

ruleLabelPropertyRef_start(scope,attr) ::= "(<scope>!=null?<labelType>(<scope>.start):null)"
ruleLabelPropertyRef_stop(scope,attr) ::= "(<scope>!=null?<labelType>(<scope>.stop):null)"
ruleLabelPropertyRef_tree(scope,attr) ::= "(<scope>!=null?<ASTLabelType>(<scope>.tree):null)"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
(<scope>!=null?(input.tokenStream.toStringWithRange(
input.treeAdaptor.getTokenStartIndex(<scope>.start),
input.treeAdaptor.getTokenStopIndex(<scope>.start))):null)
<else>
(<scope>!=null?input.toStringWithTokenRange(<scope>.start,<scope>.stop):null)
<endif>
>>

ruleLabelPropertyRef_st(scope,attr) ::= "(<scope>!=null?<scope>.st:null)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

```

```

lexerRuleLabelPropertyRef_type(scope,attr) ::=
    "<scope>!=null?<scope>.type:0)"
lexerRuleLabelPropertyRef_line(scope,attr) ::=
    "<scope>!=null?<scope>.lien:0)"
lexerRuleLabelPropertyRef_pos(scope,attr) ::=
    "<scope>!=null?<scope>.charPositionInLine:0)"
lexerRuleLabelPropertyRef_channel(scope,attr) ::=
    "<scope>!=null?<scope>.channel:0)"
lexerRuleLabelPropertyRef_index(scope,attr) ::=
    "<scope>!=null?<scope>.tokenIndex:0)"
lexerRuleLabelPropertyRef_text(scope,attr) ::=
    "<scope>!=null?<scope>.text:null)"
lexerRuleLabelPropertyRef_int(scope,attr) ::=
    "<scope>!=null?int(<scope>.text):0)"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "<labelType>(retval.start)"
rulePropertyRef_stop(scope,attr) ::= "<labelType>(retval.stop)"
rulePropertyRef_tree(scope,attr) ::= "<ASTLabelType>(retval.tree)"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
input.tokenStream.toStringWithRange(
    input.treeAdaptor.getTokenStartIndex(retval.start),
    input.treeAdaptor.getTokenStopIndex(retval.start))
<else>
input.toStringWithTokenRange(retval.start,input.LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "text"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(charIndex-1)"
lexerRulePropertyRef_int(scope,attr) ::= "int(<scope>.text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>;"

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>

```

```

if ( <actions.(actionScope).synpredgate> ) {
  <action>
}
<else>
<action>
<endif>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
public static const <name>:BitSet = new BitSet([<words64:{it |<it>};separator=", ">]);<n>
>>

codeFileExtension() ::= ".as"

true_value() ::= "true"
false_value() ::= "false"

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ActionScript/ActionScript.stg

```

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2005-2006 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** The API version of the runtime that recognizers generated by this runtime

* need.

*/

apiVersion() ::= "1"

// System.Boolean.ToString() returns "True" and "False", but the proper C# literals are "true" and "false"

// The Java version of Boolean returns "true" and "false", so they map to themselves here.

booleanLiteral ::= [

 "True": "true",

 "False": "false",

 "true": "true",

 "false": "false",

 default: "false"

]

/** The overall file structure of a recognizer; stores methods for rules

* and cyclic DFAs plus support code.

*/

outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,

 docComment, recognizer,

 name, tokens, tokenNames, rules, cyclicDFAs,

 bitsets, buildTemplate, buildAST, rewriteMode, profile,

 backtracking, synpreds, memoize, numRules,

 fileName, ANTLRVersion, generatedTimestamp, trace,

 scopes, superClass, literals) ::=

<<

\$ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>

<@imports>

import sys

from antlr3 import *

<if(TREE_PARSER)>

from antlr3.tree import *<\n>

<endif>

from antlr3.compat import set, frozenset

<@end>

<actions.(actionScope).header>

<! <docComment> !>


```

# for convenience in actions
HIDDEN = BaseRecognizer.HIDDEN

# token types
<tokens:{it | <it.name>=<it.type>}; separator="\n">

<recognizer>

<if(actions.(actionScope).main)>
<actions.(actionScope).main>
<else>
def main(argv, stdin=sys.stdin, stdout=sys.stdout, stderr=sys.stderr):
<if(LEXER)>
    from antlr3.main import LexerMain
    main = LexerMain(<recognizer.name>)<\n>
<endif>
<if(PARSER)>
    from antlr3.main import ParserMain
    main = ParserMain("<recognizer.grammar.name>Lexer", <recognizer.name>)<\n>
<endif>
<if(TREE_PARSER)>
    from antlr3.main import WalkerMain
    main = WalkerMain(<recognizer.name>)<\n>
<endif>
    main.stdin = stdin
    main.stdout = stdout
    main.stderr = stderr
    main.execute(argv)<\n>
<endif>

<actions.(actionScope).footer>

if __name__ == '__main__':
    main(sys.argv)

>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode,
    labelText="CommonToken", superClass="Lexer") ::= <<
<grammar.directDelegates:
{g|from <g.recognizerName> import <g.recognizerName>}; separator="\n">

class <grammar.recognizerName>(<@superClassName><superClass><@end>):
    <scopes:{it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>

    grammarFileName = "<fileName>"
    api_version = <apiVersion()>

```

```

def __init__(self<grammar.delegators:{g|,<g:delegateName(>>}, input=None, state=None):
    if state is None:
        state = RecognizerSharedState()
        super(<grammar.recognizerName>, self).__init__(input, state)

<if(memoize)>
<if(grammar.grammarIsRoot)>
    self._state.ruleMemo = {}
<endif>
<endif>

    <grammar.directDelegates:
        {g|self.<g:delegateName(> = <g.recognizerName>(<trunc(g.delegators):{p|<p:delegateName(>, }>self, input,
state)); separator="\n">
    <grammar.directDelegates:
        {g|<g.delegates:{h|self.<h:delegateName(> = self.<g:delegateName(>.<h:delegateName(>};
separator="\n">}; separator="\n">
    <grammar.delegators:
        {g|self.<g:delegateName(> = <g:delegateName(>}; separator="\n">
    <last(grammar.delegators):
        {g|self.gParent = <g:delegateName(>}; separator="\n">
        self.delegates = [<grammar.delegates: {g|self.<g:delegateName(>}; separator = ", ">]

    <cyclicDFAs:{dfa | <cyclicDFAInit(dfa)>}; separator="\n">

    <actions.lexer.init>

    <actions.lexer.members>

<if(filterMode)>
    <filteringNextToken(>
<endif>
    <rules; separator="\n\n">

    <synpreds:{p | <lexerSynpred(p)>}>

    <cyclicDFAs:cyclicDFA(> <! dump tables for all DFA !>

>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
* for matches. No error can be generated upon error; just rewind, consume
* a token and then try again. backtracking needs to be set as well.
* Make rule memoization happen only at levels above 1 as we start mTokens
* at backtracking==1.

```

```

*/
filteringNextToken() ::= <<
def nextToken(self):
    while True:
        if self.input.LA(1) == EOF:
            return self.makeEOFToken()

        self._state.token = None
        self._state.channel = DEFAULT_CHANNEL
        self._state.tokenStartCharIndex = self.input.index()
        self._state.tokenStartCharPositionInLine = self.input.charPositionInLine
        self._state.tokenStartLine = self.input.line
        self._state._text = None
        try:
            m = self.input.mark()
            try:
                # means we won't throw slow exception
                self._state.backtracking = 1
                try:
                    self.mTokens()
                finally:
                    self._state.backtracking = 0

            except BacktrackingFailed:
                # mTokens backtracks with synpred at backtracking==2
                # and we set the synpredgate to allow actions at level 1.
                self.input.rewind(m)
                self.input.consume() # advance one char and try again

        else:
            self.emit()
            return self._state.token

    except RecognitionException, re:
        # shouldn't happen in backtracking mode, but...
        self.reportError(re)
        self.recover(re)

def memoize(self, input, ruleIndex, ruleStartIndex, success):
    if self._state.backtracking > 1:
        # is Lexer always superclass?
        super(<grammar.recognizerName>, self).memoize(input, ruleIndex, ruleStartIndex, success)

def alreadyParsedRule(self, input, ruleIndex):
    if self._state.backtracking > 1:
        return super(<grammar.recognizerName>, self).alreadyParsedRule(input, ruleIndex)

```

```

return False

>>

actionGate() ::= "self._state.backtracking == 0"

filteringActionGate() ::= "self._state.backtracking == 1"

/** How to generate a parser */

genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
              bitsets, inputStreamType, superClass, labelType, members,
              rewriteElementType, filterMode, init, ASTLabelType="Object") ::= <<
<if(grammar.grammarIsRoot)>
# token names
tokenNames = [
    "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>",
    <tokenNames; wrap, separator=", ">
]<\n>
<else>
from <grammar.composite.rootGrammar.recognizerName> import tokenNames<\n>
<endif>
<scopes:{it|<if(it.isDynamicGlobalScope)><globalAttributeScopeClass(scope=it)><endif>}>

<grammar.directDelegates:
{g|from <g.recognizerName> import <g.recognizerName>; separator="\n">

<rules:{it|<ruleAttributeScopeClass(scope=it.ruleDescriptor.ruleScope)>}>

class <grammar.recognizerName>(<@superClassName><superClass><@end>):
    grammarFileName = "<fileName>"
    api_version = <apiVersion()>
    tokenNames = tokenNames

    def __init__(self<grammar.delegators:{g|, <g:delegateName()>}>, input, state=None, *args, **kwargs):
        if state is None:
            state = RecognizerSharedState()

        <@args()>
        super(<grammar.recognizerName>, self).__init__(input, state, *args, **kwargs)

<if(memoize)>
<if(grammar.grammarIsRoot)>
    self._state.ruleMemo = {}
<endif>
<endif>

```

```

<cyclicDFAs:{ dfa | <cyclicDFAInit(dfa)>}; separator="\n">

<scopes:{ it | <if(it.isDynamicGlobalScope)><globalAttributeScopeStack(scope=it)><endif> }>
<rules:{ it | <ruleAttributeScopeStack(scope=it.ruleDescriptor.ruleScope)> }>

<init>

<grammar.delegators:
  {g|self.<g:delegateName()> = <g:delegateName()>}; separator="\n">
<grammar.directDelegates:
  {g|self.<g:delegateName()> = <g.recognizerName>(<trunc(g.delegators):{p|<p:delegateName()>, }>self, input,
state)); separator="\n">
  <grammar.directDelegates:
    {g|<g.delegates:{h|self.<h:delegateName()> = self.<g:delegateName()>.<h:delegateName()>};
separator="\n">}; separator="\n">
  <last(grammar.delegators):
    {g|self.gParent = self.<g:delegateName()>}; separator="\n">
    self.delegates = [<grammar.delegates: {g|self.<g:delegateName()>}; separator = ", ">]

<@init><@end>

<@members><@end>

<members>

<rules; separator="\n\n">

<! generate rule/method definitions for imported rules so they
  appear to be defined in this recognizer. !>
<grammar.delegatedRules:{ ruleDescriptor| <delegateRule(ruleDescriptor)> }; separator="\n">

<synpreds:{ p | <synpred(p)> }>

<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

<bitsets:{ it | FOLLOW_<it.name>_in_<it.inName><it.tokenIndex> = frozenset([<it.tokenTypes:{ it |
<it>};separator=", ">])<n> }>

>>

delegateRule(ruleDescriptor) ::= <<
def <ruleDescriptor.name>(self, <ruleDescriptor.parameterScope:parameterScope()>):
<\> <if(ruleDescriptor.hasReturnValue)>return
<endif>self.<ruleDescriptor.grammar:delegateName()>.<ruleDescriptor.name>(<if(ruleDescriptor.parameterScope)
><ruleDescriptor.parameterScope.attributes:{a|<a.name>}; separator=", "><endif>)

```

```
>>
```

```
parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,  
  ASTLabelType="Object", superClass="Parser", labelType="Token",  
  members={<actions.parser.members>},  
  init={<actions.parser.init>}  
  ) ::= <<
```

```
<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,  
  bitsets, "TokenStream", superClass,  
  labelType, members, "Token",  
  false, init, ASTLabelType)>
```

```
>>
```

```
/** How to generate a tree parser; same as parser except the input
```

```
* stream is a different type.
```

```
*/
```

```
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,  
  numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="Object",  
  superClass={<if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif>},  
  members={<actions.treeparser.members>},  
  init={<actions.treeparser.init>}  
  ) ::= <<
```

```
<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,  
  bitsets, "TreeNodeStream", superClass,  
  labelType, members, "Node",  
  filterMode, init, ASTLabelType)>
```

```
>>
```

```
/** A simpler version of a rule template that is specific to the imaginary
```

```
* rules created for syntactic predicates. As they never have return values
```

```
* nor parameters etc..., just give simplest possible method. Don't do
```

```
* any of the normal memoization stuff in here either; it's a waste.
```

```
* As predicates cannot be inlined into the invoking rule, they need to
```

```
* be in a rule by themselves.
```

```
*/
```

```
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
```

```
<<
```

```
# $ANTLR start "<ruleName>"
```

```
def <ruleName>_fragment(self, <ruleDescriptor.parameterScope:parameterScope()):
```

```
  <ruleLabelDefs>
```

```
<if(trace)>
```

```
  self.traceIn("<ruleName>_fragment", <ruleDescriptor.index>)
```

```
  try:
```

```
    <block>
```

```
  finally:
```

```
    self.traceOut("<ruleName>_fragment", <ruleDescriptor.index>)
```

```
<else>
  <block>
<endif>
# $ANTLR end "<ruleName>"
```

```
>>
```

```
synpred(name) ::= <<
def <name>(self):
  self._state.backtracking += 1
  <@start()>
  start = self.input.mark()
  try:
    self.<name>_fragment()
  except BacktrackingFailed:
    success = False
  else:
    success = True
  self.input.rewind(start)
  <@stop()>
  self._state.backtracking -= 1
  return success
```

```
>>
```

```
lexerSynpred(name) ::= <<
<synpred(name)>
>>
```

```
ruleMemoization(name) ::= <<
<if(memoize)>
if self._state.backtracking > 0 and self.alreadyParsedRule(self.input, <ruleDescriptor.index>):
  # for cached failed rules, alreadyParsedRule will raise an exception
  success = True
  return <ruleReturnValue()>
```

```
<endif>
>>
```

```
/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>
if self._state.backtracking > 0:
  raise BacktrackingFailed

<endif>
```

```
>>
```

```
/** How to generate code for a rule. This includes any return type
```

```
* data aggregates required for multiple return values.
```

```
*/
```

```
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
```

```
<returnScope(scope=ruleDescriptor.returnScope)>
```

```
# $ANTLR start "<ruleName>"
```

```
# <fileName>:<description>
```

```
<ruleDescriptor.actions.decorate>
```

```
def <ruleName>(self, <ruleDescriptor.parameterScope:parameterScope()>):
```

```
<if(trace)>
```

```
    self.traceIn("<ruleName>", <ruleDescriptor.index>)<\n>
```

```
<endif>
```

```
    <ruleScopeSetUp()>
```

```
    <ruleDeclarations()>
```

```
    <ruleLabelDefs()>
```

```
    <ruleDescriptor.actions.init>
```

```
    <@preamble()>
```

```
    <@body><ruleBody()><@end>
```

```
    <@postamble()>
```

```
    return <ruleReturnValue()>
```

```
# $ANTLR end "<ruleName>"
```

```
>>
```

```
ruleBody() ::= <<
```

```
<if(memoize)>
```

```
<if(backtracking)>
```

```
success = False<\n>
```

```
<endif>
```

```
<endif>
```

```
try:
```

```
    try:
```

```
        <ruleMemoization(name=ruleName)>
```

```
        <block>
```

```
        <ruleCleanUp()>
```

```
        <(ruleDescriptor.actions.after):execAction()>
```

```
<if(memoize)>
```

```
<if(backtracking)>
```

```
    success = True<\n>
```

```
<endif>
```

```
<endif>
```

```
<if(exceptions)>
```

```
    <exceptions:{e|<catch(decl=e.decl,action=e.action)><\n>}>
```

```
<else>
```



```

<if(!emptyRule)>
<if(actions.(actionScope).rulecatch)>
  <actions.(actionScope).rulecatch>
<else>
  except RecognitionException, re:
    self.reportError(re)
    self.recover(self.input, re)
    <@setErrorReturnValue()>

<endif>
<else>
  finally:
    pass

<endif>
<endif>
finally:
<if(trace)>
  self.traceOut("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
  <memoize()>
  <ruleScopeCleanUp()>
  <finally>
  pass
>>

catch(decl,action) ::= <<
except <e.decl>:
  <e.action>

>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval = self.<ruleDescriptor.name>_return()
retval.start = self.input.LT(1)<\n>
<elseif(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes: { a |
<a.name> = <if(a.initValue)><a.initValue><else>None<endif>
}>
<endif>
<if(memoize)>
<ruleDescriptor.name>_startIndex = self.input.index()
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: {it | self.<it>_stack.append(<it>_scope())}; separator="\n">

```

```

<ruleDescriptor.ruleScope:{it | self.<it.name>_stack.append(<it.name>_scope()); separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes:{it | self.<it>_stack.pop()}; separator="\n">
<ruleDescriptor.ruleScope:{it | self.<it.name>_stack.pop()}; separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it | <it.label.text> = None }; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,
ruleDescriptor.wildcardTreeListLabels]
: {it | list_<it.label.text> = None }; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels:{it | <it.label.text> = None }; separator="\n">
>>

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | <it.label.text> = None }; separator="\n"
>
<ruleDescriptor.charLabels:{it | <it.label.text> = None }; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]
: {it | list_<it.label.text> = None }; separator="\n"
>
>>

ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
<endif>
%>

ruleCleanUp() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>

```

```

<if(!TREE_PARSER)>
retval.stop = self.input.LT(-1)<\n>
<endif>
<endif>
>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if self._state.backtracking > 0:
    self.memoize(self.input, <ruleDescriptor.index>, <ruleDescriptor.name>_startIndex, success)

<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
# $ANTLR start "<ruleName>"
def m<ruleName>(self, <ruleDescriptor.parameterScope:parameterScope()>):
<if(trace)>
    self.traceIn("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
<if(memoize)>
<if(backtracking)>
    success = False<\n>
<endif>
<endif>
    try:
<if(nakedBlock)>
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block><\n>
<else>
        _type = <ruleName>
        _channel = DEFAULT_CHANNEL

        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block>
        <ruleCleanUp()>
        self._state.type = _type

```

```

        self._state.channel = _channel
        <(ruleDescriptor.actions.after):execAction(>
<endif>
<if(memoize)>
<if(backtracking)>
    success = True<\n>
<endif>
<endif>

    finally:
<if(trace)>
    self.traceOut("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
<ruleScopeCleanUp(>
    <memoize(>
    pass

# $ANTLR end "<ruleName>"

>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
def mTokens(self):
    <block><\n>

>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
alt<decisionNumber> = <maxAlt>
<decls>
<@body><blockBody()><@end>
>>

blockBody() ::= <<
<@predecision()>
<@decision><decision><@end>
<@postdecision()>
<@prebranch()>
<alts:{ a | <altSwitchCase(i, a)> }; separator="\n" >

```

```

<@postbranch(>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
alt<decisionNumber> = <maxAlt>
<decls>
<@predecision(>
<@decision><decision><@end>
<@postdecision(>
<alts:{a | <altSwitchCase(i, a)>}; separator="\nel">
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
cnt<decisionNumber> = 0
<decls>
<@preloop(>
<@loopBody>
<positiveClosureBlockLoop(>
<@end>
<@postloop(>
>>

positiveClosureBlockLoop() ::= <<
while True: #loop<decisionNumber>
  alt<decisionNumber> = <maxAlt>
  <@predecision(>

```

```

<@decisionBody><decision><@end>
<@postdecision()>
<alts:{a | <altSwitchCase(i, a)>}; separator="\nel">
else:
  if cnt<decisionNumber> >= 1:
    break #loop<decisionNumber>

  <ruleBacktrackFailure()>
  eee = EarlyExitException(<decisionNumber>, self.input)
  <@earlyExitException()>
  raise eee

  cnt<decisionNumber> += 1
>>

positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
# <fileName>:<description>
<decls>
<@preloop()>
<@loopBody>
<closureBlockLoop()>
<@end>
<@postloop()>
>>

closureBlockLoop() ::= <<
while True: #loop<decisionNumber>
  alt<decisionNumber> = <maxAlt>
  <@predecision()>
  <@decisionBody><decision><@end>
  <@postdecision()>
  <alts:{a | <altSwitchCase(i, a)>}; separator="\nel">
  else:
    break #loop<decisionNumber>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation
* so we can just use the normal block template
*/
optionalBlock ::= block

optionalBlockSingleAlt ::= block

```

```

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */
altSwitchCase(altNum,alt) ::= <<
if alt<decisionNumber> == <altNum>:
  <@prealt()>
  <alt>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt, treeLevel,rew) ::= <<
# <fileName>:<description>
pass <! so empty alternatives are a valid block !>
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
>>

/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(e) ::= <<
<@prematch()>
<e.el><\n>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<if(label)><label> = <endif>self.match(self.input, <token>,
self.FOLLOW_<token>_in_<ruleName><elementIndex>)
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(token,label,elementIndex,terminalOptions)>
<listLabel(label, label)>
>>

listLabel(label, elem) ::= <<
if list_<label> is None:

```

```

    list_<label> = []
list_<label>.append(<elem>)<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = self.input.LA(1)<\n>
<endif>
self.match(<char>)
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = self.input.LA(1)<\n>
<endif>
self.matchRange(<a>, <b>)
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="",terminalOptions={}) ::= <<
<if(label)>
<label> = self.input.LT(1)<\n>
<endif>
if <s>:
    self.input.consume()
    <postmatchCode>
<if(!LEXER)>
    self._state.errorRecovery = False<\n>
<endif>

else:
    <ruleBacktrackFailure()>
    mse = MismatchedSetException(None, self.input)
    <@mismatchedSetException()>
<if(LEXER)>
    self.recover(mse)
    raise mse
<else>
    raise mse
<! use following code to make it recover inline; remove throw mse;
self.recoverFromMismatchedSet(
    self.input, mse, self.FOLLOW_set_in_<ruleName><elementIndex>
)
!>
<endif>
<\n>

```



```
>>
```

```
matchRuleBlockSet ::= matchSet
```

```
matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
```

```
<matchSet(...)>
```

```
<listLabel(label, label)>
```

```
>>
```

```
/** Match a string literal */
```

```
lexerStringRef(string,label,elementIndex="0") ::= <<
```

```
<if(label)>
```

```
<label>Start = self.getCharIndex()
```

```
self.match(<string>)
```

```
<label>StartLine<elementIndex> = self.getLine()
```

```
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
```

```
<label> = <labelType>(input=self.input, type=INVALID_TOKEN_TYPE, channel=DEFAULT_CHANNEL,  
start=<label>Start, stop=self.getCharIndex()-1)
```

```
<label>.setLine(<label>StartLine<elementIndex>)
```

```
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
```

```
<else>
```

```
self.match(<string>)
```

```
<endif>
```

```
>>
```

```
wildcard(token,label,elementIndex,terminalOptions={}) ::= <<
```

```
<if(label)>
```

```
<label> = self.input.LT(1)<\n>
```

```
<endif>
```

```
self.matchAny(self.input)
```

```
>>
```

```
wildcardAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
```

```
<wildcard(...)>
```

```
<listLabel(label,label)>
```

```
>>
```

```
/** Match . wildcard in lexer */
```

```
wildcardChar(label, elementIndex) ::= <<
```

```
<if(label)>
```

```
<label> = self.input.LA(1)<\n>
```

```
<endif>
```

```
self.matchAny()
```

```
>>
```

```
wildcardCharListLabel(label, elementIndex) ::= <<
```

```
<wildcardChar(label, elementIndex)>
```

```
<listLabel(label, label)>
```

```
>>
```

```
/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
self._state.following.append(self.FOLLOW_<rule.name>_in_<ruleName><elementIndex>)
<if(label)><label> = <endif>self.<if(scope)><scope:delegateName().<endif><rule.name>(<args; separator=",
">><\n>
self._state.following.pop()
>>
```

```
/** ids+=rule */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, label)>
>>
```

```
/** A lexer rule reference
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
<label>Start<elementIndex> = self.getCharIndex()
self.<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">)
<label>StartLine<elementIndex> = self.getLine()
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
<label> = <labelType>(
input=self.input,
type=INVALID_TOKEN_TYPE,
channel=DEFAULT_CHANNEL,
start=<label>Start<elementIndex>,
stop=self.getCharIndex()-1)
<label>.setLine(<label>StartLine<elementIndex>)
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
<else>
self.<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">)
<endif>
>>
```

```
/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(rule,label,args,elementIndex,scope)>
```

```

<listLabel(label, label)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
<label>Start<elementIndex> = self.getCharIndex()
<label>StartLine<elementIndex> = self.getLine()
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
self.match(EOF)
<label> = <labelType>(input=self.input, type=EOF, channel=DEFAULT_CHANNEL,
start=<label>Start<elementIndex>, stop=self.getCharIndex()-1)
<label>.setLine(<label>StartLine<elementIndex>)
<label>.setCharPositionInLine(<label>StartCharPos<elementIndex>)
<else>
self.match(EOF)
<endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "<recRuleArg()>"
recRuleArg()            ::= "_p"
recRuleAltPredicate(ruleName, opPrec) ::= "<recRuleArg()> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0 = $<ruleName>_primary.tree"
recRuleSetReturnAction(src, name)    ::= "$<name> = $<src>.<name>"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if self.input.LA(1) == DOWN:
self.match(self.input, DOWN, None)
<children:element()>
self.match(self.input, UP, None)

<else>
self.match(self.input, DOWN, None)
<children:element()>
self.match(self.input, UP, None)
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
* also hoisted into a prediction expression).
*/
validateSemanticPredicate(pred,description) ::= <<

```

```

if not (<evalPredicate(pred, description)>):
    <ruleBacktrackFailure()>
    raise FailedPredicateException(self.input, "<ruleName>", "<description>")

>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n" >
else:
<if(eotPredictsAlt)>
    alt<decisionNumber> = <eotPredictsAlt>
<else>
    <ruleBacktrackFailure()>
    nvae = NoViableAltException("<description>", <decisionNumber>, <stateNumber>, self.input)<\n>
    <@noViableAltException()>
    raise nvae<\n>
<endif>
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n" >
>>

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n" ><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber> = <eotPredictsAlt> <! if no edges, don't gen ELSE !>
<else>
else:
    alt<decisionNumber> = <eotPredictsAlt>
<\n>

```

```

<endif>
<endif>
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber> = <alt>"

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if (<labelExpr>) <if(predicates)>and (<predicates>)<endif>:
    <targetState>
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
<!
    FIXME: this is one of the few occasion, where I miss a switch statement
    in Python. ATM this is implemented as a list of if .. elif ..
    This may be replaced by faster a dictionary lookup, when I find a solution
    for the cases when an edge is not a plain dfaAcceptState.
!>
LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n">
else:
<if(eotPredictsAlt)>
    alt<decisionNumber> = <eotPredictsAlt>
<else>
    <ruleBacktrackFailure()>
    nvae = NoViableAltException("<description>", <decisionNumber>, <stateNumber>, self.input)<\n>
    <@noViableAltException()>
    raise nvae<\n>
<endif>

>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n">
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<

```

```

LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n\n">
<if(eotPredictsAlt)>
else:
    alt<decisionNumber> = <eotPredictsAlt>
<endif>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
if <labels:{it | LA<decisionNumber> == <it>} ; separator=" or ">:
    <targetState>
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = self.dfa<decisionNumber>.predict(self.input)
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
# lookup tables for DFA #<dfa.decisionNumber>

DFA<dfa.decisionNumber>_eot = DFA.unpack(
    u"<dfa.javaCompressedEOT; wrap="\n    u\">"
)

DFA<dfa.decisionNumber>_eof = DFA.unpack(
    u"<dfa.javaCompressedEOF; wrap="\n    u\">"
)

DFA<dfa.decisionNumber>_min = DFA.unpack(
    u"<dfa.javaCompressedMin; wrap="\n    u\">"
)

DFA<dfa.decisionNumber>_max = DFA.unpack(
    u"<dfa.javaCompressedMax; wrap="\n    u\">"
)

```

```

)

DFA<dfa.decisionNumber>_accept = DFA.unpack(
  u"<dfa.javaCompressedAccept; wrap=""\n  u"">"
)

DFA<dfa.decisionNumber>_special = DFA.unpack(
  u"<dfa.javaCompressedSpecial; wrap=""\n  u"">"
)

DFA<dfa.decisionNumber>_transition = [
  <dfa.javaCompressedTransition:{s|DFA.unpack(u"<s; wrap=""\nu"">)}; separator=",\n">
]

# class definition for DFA #<dfa.decisionNumber>

class DFA<dfa.decisionNumber>(DFA):
  pass

  <@errorMethod()>

  <if(dfa.specialStateSTs)>
  def specialStateTransition(self_, s, input):
    # convince pylint that my self_ magic is ok ;)
    # pylint: disable-msg=E0213

    # pretend we are a member of the recognizer
    # thus semantic predicates can be evaluated
    self = self_.recognizer

    _s = s

    <dfa.specialStateSTs:{state | if s == <i0>: <! compressed special state numbers 0..n-1 !>
    <state>}; separator="\nel">

  <if(backtracking)>
    if self._state.backtracking > 0:
      raise BacktrackingFailed

  <endif>
    nvae = NoViableAltException(self_.getDescription(), <dfa.decisionNumber>, _s, input)
    self_.error(nvae)
    raise nvae<\n>
  <endif>

  >>

```

```

cyclicDFAInit(dfa) ::= <<
self.dfa<dfa.decisionNumber> = self.DFA<dfa.decisionNumber>(
  self, <dfa.decisionNumber>,
  eot = self.DFA<dfa.decisionNumber>_eot,
  eof = self.DFA<dfa.decisionNumber>_eof,
  min = self.DFA<dfa.decisionNumber>_min,
  max = self.DFA<dfa.decisionNumber>_max,
  accept = self.DFA<dfa.decisionNumber>_accept,
  special = self.DFA<dfa.decisionNumber>_special,
  transition = self.DFA<dfa.decisionNumber>_transition
)<\n>
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = input.LA(1)<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
index<decisionNumber>_<stateNumber> = input.index()
input.rewind()<\n>
<endif>
s = -1
<edges; separator="\n">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>)<\n>
<endif>
if s >= 0:
  return s
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if (<labelExpr>)<if(predicates)> and (<predicates>)<endif>:
  s = <targetStateNumber><\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
se:
  s = <targetStateNumber><\n>
>>

```


// D F A E X P R E S S I O N S

```
andPredicates(left,right) ::= "((<left>) and (<right>))"
```

```
orPredicates(operands) ::= "<operands; separator=\" or \">"
```

```
notPredicate(pred) ::= "not (<evalPredicate(pred, {}>)"
```

```
evalPredicate(pred,description) ::= "<pred>"
```

```
evalSynPredicate(pred,description) ::= "self.<pred>()"
```

```
lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber> == <atom>"
```

```
/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
```

```
* somewhere. Must ask for the lookahead directly.
```

```
*/
```

```
isolatedLookaheadTest(atom,k,atomAsInt) ::= "self.input.LA(<k>) == <atom>"
```

```
lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
```

```
<lower> \<= LA<decisionNumber>_<stateNumber> \<= <upper>
```

```
%>
```

```
isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "<lower> \<= self.input.LA(<k>) \<= <upper>"
```

```
setTest(ranges) ::= "<ranges; separator=\" or \">"
```

// A T T R I B U T E S

```
globalAttributeScopeClass(scope) ::= <<
```

```
<if(scope)>
```

```
<if(scope.attributes)>
```

```
class <scope.name>_scope(object):
```

```
    def __init__(self):
```

```
        <scope.attributes:{it | self.<it.decl> = None}; separator="\n">
```

```
<endif>
```

```
<endif>
```

```
>>
```

```
globalAttributeScopeStack(scope) ::= <<
```

```
<if(scope)>
```

```
<if(scope.attributes)>
```

```
self.<scope.name>_stack = []<\n>
```

```
<endif>
```

```
<endif>
```

```
>>
```

```

ruleAttributeScopeClass(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
class <scope.name>_scope(object):
    def __init__(self):
        <scope.attributes:{it | self.<it.decl> = None}; separator="\n">

<endif>
<endif>
>>

ruleAttributeScopeStack(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
self.<scope.name>_stack = []<\n>
<endif>
<endif>
>>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
<label.label.text> = None
>>

returnStructName(r) ::= "<r.name>_return"

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
class <ruleDescriptor:returnStructName()>(<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope):
    def __init__(self):
        super(<grammar.recognizerName>.<ruleDescriptor:returnStructName()>, self).__init__()

        <if(scope)><scope.attributes:{it | self.<it.decl> = None}; separator="\n"><endif>
        <@ruleReturnInit()>

        <@ruleReturnMembers()>

<endif>

```

>>

```
parameterScope(scope) ::= <<  
<if(scope)><scope.attributes:{it | <it.decl>}; separator=", "><endif>  
>>
```

```
parameterAttributeRef(attr) ::= "<attr.name>"  
parameterSetAttributeRef(attr,expr) ::= "<attr.name> = <expr>"
```

```
scopeAttributeRef(scope,attr,index,negIndex) ::= <%  
<if(negIndex)>  
self.<scope>_stack[-<negIndex>].<attr.name>  
<else>  
<if(index)>  
self.<scope>_stack[<index>].<attr.name>  
<else>  
self.<scope>_stack[-1].<attr.name>  
<endif>  
<endif>  
>%
```

/* not applying patch because of bug in action parser!

```
<if(negIndex)>  
((len(self.<scope>_stack) - <negIndex> - 1) >= 0 and [self.<scope>_stack[-<negIndex>].<attr.name>] or [None])[0]  
<else>  
<if(index)>  
((<index> \< len(self.<scope>_stack)) and [self.<scope>_stack[<index>].<attr.name>] or [None])[0]  
<else>  
((len(self.<scope>_stack) > 0) and [self.<scope>_stack[-1].<attr.name>] or [None])[0]  
<endif>  
<endif>
```

*/

```
scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%  
<if(negIndex)>  
<!FIXME: this seems not to be used by ActionTranslator...!>  
self.<scope>_stack[-<negIndex>].<attr.name> = <expr>  
<else>  
<if(index)>  
<!FIXME: this seems not to be used by ActionTranslator...!>  
self.<scope>_stack[<index>].<attr.name> = <expr>  
<else>  
self.<scope>_stack[-1].<attr.name> = <expr>  
<endif>  
<endif>  
>%
```

```

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size()>0 && $function::name.equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "self.<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
((<scope> is not None) and [<scope>.<attr.name>] or [None])[0]
<else>
<scope>
<endif>
%>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
%>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> = <expr>
<else>
<attr.name> = <expr>
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach; and they are evaluated early;
// they cannot see TREE_PARSER or PARSER attributes for example. :(

tokenLabelPropertyRef_text(scope,attr) ::= "<scope>.text"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>.type"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>.line"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>.charPositionInLine"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>.channel"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>.index"

```

```

tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>.start"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>.stop"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>.tree"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
((<scope> is not None) and [self.input.getTokenStream().toString(
    self.input.getTreeAdaptor().getTokenStartIndex(<scope>.start),
    self.input.getTreeAdaptor().getTokenStopIndex(<scope>.start)
    ]) or [None]][0]
<else>
((<scope> is not None) and [self.input.toString(<scope>.start,<scope>.stop)] or [None])[0]
<endif>
%>
ruleLabelPropertyRef_st(scope,attr) ::= "((<scope> is not None) and [<scope>.st] or [None])[0]"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "((<scope> is not None) and [<scope>.type] or [0])[0]"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "((<scope> is not None) and [<scope>.line] or [0])[0]"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "((<scope> is not None) and [<scope>.charPositionInLine] or [0])[0]"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "((<scope> is not None) and [<scope>.channel] or [0])[0]"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "((<scope> is not None) and [<scope>.index] or [0])[0]"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "((<scope> is not None) and [<scope>.text] or [None])[0]"
lexerRuleLabelPropertyRef_int(scope,attr) ::= "((<scope> is not None) and [int(<scope>.text)] or [0])[0]"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.start"
rulePropertyRef_stop(scope,attr) ::= "retval.stop" //mmm... or input.LT(-1)??
rulePropertyRef_tree(scope,attr) ::= "retval.tree"
rulePropertyRef_text(scope,attr) ::= "self.input.toString(retval.start, self.input.LT(-1))"
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "self.text"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "self._state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "self._state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "self._state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(self.getCharIndex()-1)"
lexerRulePropertyRef_int(scope,attr) ::= "int(<scope>.text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error

```

```
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>"
```

```
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>"
```

```
/** How to execute an action (only when not backtracking) */
```

```
execAction(action) ::= <<
```

```
<if(backtracking)>
```

```
<if(actions.(actionScope).synpredgate)>
```

```
if <actions.(actionScope).synpredgate>:
```

```
    pass
```

```
    <action>
```

```
<else>
```

```
if <actions.(actionScope).synpredgate>:
```

```
    pass
```

```
    <action>
```

```
<endif>
```

```
<else>
```

```
#action start
```

```
<action>
```

```
#action end
```

```
<endif>
```

```
>>
```

```
/** How to always execute an action even when backtracking */
```

```
execForcedAction(action) ::= "<action>"
```

```
// M I S C (properties, etc...)
```

```
codeFileExtension() ::= ".py"
```

```
true_value() ::= "True"
```

```
false_value() ::= "False"
```

```
Found in path(s):
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/Python/Python.stg
```

```
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2012 Terence Parr
```

```
All rights reserved.
```

```
Redistribution and use in source and binary forms, with or without  
modification, are permitted provided that the following conditions
```

are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to AST stuff. Dynamic inheritance

* hierarchy is set up as ASTDbg : AST : Dbg : Python by code generator.

*/

group ASTDbg;

astAccessor() ::= <<

def setTreeAdaptor(self, adaptor):

<if(grammar.grammarIsRoot)>

self._adaptor = DebugTreeAdaptor(self.dbg, adaptor)

<else>

self._adaptor = adaptor # delegator sends dbg adaptor

<endif>

<grammar.directDelegates:{g|<g.delegateName()>.setTreeAdaptor(self._adaptor)}>

def getTreeAdaptor(self):

return self._adaptor

adaptor = property(getTreeAdaptor, setTreeAdaptor)<\n>

>>

createListenerAndHandshake() ::= <<

proxy = DebugEventSocketProxy(self,

adaptor=<if(TREE_PARSER)>self.input.getTreeAdaptor()<else>self._adaptor<endif>,

debug=debug_socket, port=port)

self.setDebugListener(proxy)

self.adaptor.setDebugListener(proxy)

self.input.setDebugListener(proxy)

```
#self.set<InputStreamType>(Debug<InputStreamType>(self.input, proxy))
proxy.handshake()
>>
```

```
@rewriteElement.pregen() ::= "self._dbg.location(<e.line>, <e.pos>)"
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python3/ASTDbg.stg
```

No license file was found, but licenses were detected in source scan.

```
/* [The "BSD license"]
```

```
Copyright (c) 2008 Erik van Bilzen
```

```
Copyright (c) 2007-2008 Johannes Luber
```

```
Copyright (c) 2005-2007 Kunle Odutola
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
group Delphi;
```

```
csharpTypeInitMap ::= [
```

```
"int": "0",
```

```
"uint": "0",
```

```
"long": "0",
```

```
"ulong": "0",
```

```
"float": "0.0",
```



```

"double": "0.0",
"bool": "False",
"byte": "0",
"sbyte": "0",
"short": "0",
"ushort": "0",
"char": "#0",
"string": "",
"String": "",
default: "nil" // anything other than an atomic type
]

```

```

/** The overall file structure of a recognizer; stores methods for rules
 * and cyclic DFAs plus support code.
 * LEXER (Boolean): should we generate lexer code?
 * PARSER (Boolean): should we generate parser code?
 * TREE_PARSER (Boolean): should we generate tree parser code?
 * actionScope (String): 'lexer', 'parser', 'tree_parser' or custom scope
 * actions (HashMap):
 * docComment (String): document comment
 * recognizer (Object): recognizer class generator
 * name (String): name of grammar
 * tokens (HashMap<name: String, type: Integer>):
 * tokenNames:
 * rules:
 * cyclicDFAs:
 * bitsets:
 * buildTemplate (Boolean): should we generate a string template?
 * buildAST (Boolean): should we generate an AST?
 * rewriteMode (Boolean): are we rewriteing nodes?
 * profile (Boolean):
 * backtracking (Boolean): backtracking mode?
 * synpreds (): syntactic predicates
 * memoize (Boolean): should we memoize?
 * numRules (Integer): number of rules
 * fileName (String): fully qualified name of original .g file
 * ANTLRVersion (String): ANTLR version in Major.Minor.Build format
 * generatedTimestamp (String): date/time when the file is generated
 * trace (Boolean): should we trace input/output?
 * scopes:
 * superClass (String): name of base class, or empty string
 * literals:
 */
outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,
           docComment, recognizer,
           name, tokens, tokenNames, rules, cyclicDFAs,
           bitsets, buildTemplate, buildAST, rewriteMode, profile,
           backtracking, synpreds, memoize, numRules,

```

```

    fileName, ANTLRVersion, generatedTimestamp, trace,
    scopes, superClass, literals) ::=
<<
unit <name>;

{$HINTS OFF}

// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>

<actions.(actionScope).header>

interface

<@imports>
uses<\n>
<@end>
<actions.(actionScope).usesInterface>
<if(TREE_PARSER)>
    Antlr.Runtime.Tree,<\n>
<endif>
    Antlr.Runtime,
    Antlr.Runtime.Collections,
    Antlr.Runtime.Tools;

<docComment>
<recognizer>
>>

/** Generates source code for the lexer class
 * grammar (Grammar object)
 */
lexer(grammar, name, tokens, scopes, rules, numRules, labelType="Token",
    filterMode, superClass="Lexer") ::= <<
type
I<grammar.recognizerName> = interface(I<@superClassName><superClass><@end>)
end;

T<grammar.recognizerName> = class(T<@superClassName><superClass><@end>,
I<grammar.recognizerName>)
strict private
    FCnt: array [0..<grammar.numberOfDecisions>] of Byte;
    FLA: array [0..<grammar.numberOfDecisions>, 0..255] of Integer;
    FException: ERecognitionException;
    procedure InitializeCyclicDFAs;
<cyclicDFAs:cyclicDFADeclaration()>
public
    const
        <tokens:{<it.name> = <it.type>;}; separator="\n">

```

```

    <scopes: { <if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif> } >
strict private
    <actions.(actionScope).memberDeclarations>
public
    // delegates
    <grammar.delegates: { g | <g.delegateName(): I<superClass>; { <g.recognizerName> } }; separator="\n" >
public
    // delegators
    <grammar.delegators: { g | <g.delegateName(): Pointer; { <g.recognizerName> } }; separator="\n" >
    <last(grammar.delegators): { g | gParent: Pointer; { <g.recognizerName> } } >
protected
    { IBaseRecognizer }
    function GetGrammarFileName: String; override;
<if(filterMode)>
    function AlreadyParsedRule(const Input: IIntStream;
        const RuleIndex: Integer): Boolean; override;
    procedure Memoize(const Input: IIntStream; const RuleIndex,
        RuleStartIndex: Integer); override;
protected
    { ILexer }
    function NextToken: IToken; override;<\n>
<endif>
protected
    { ILexer }
    procedure DoTokens; override;
public
    constructor Create; overload;
    constructor Create(const AInput: ICharStream<grammar.delegators: { g | <g.delegateName():
IBaseRecognizer{ <g.recognizerName> } } >); overload;
    constructor Create(const AInput: ICharStream; const AState: IRecognizerSharedState<grammar.delegators: { g |
const A<g.delegateName(): IBaseRecognizer{ <g.recognizerName> } } >); overload;

    <rules: { r | <if(!r.ruleDescriptor.isSynPred)><lexerRuleDeclaration(r)><endif> } >
    <synpreds: { p | <lexerSynpredDeclaration(p)> }; separator="\n" >
end;

implementation

uses
    <grammar.delegates: { g | <g.recognizerName>; }; separator="\n" >
    <grammar.delegators: { g | <g.recognizerName>; }; separator="\n" >
    <actions.(actionScope).usesImplementation>
    SysUtils,
    StrUtils,
    Math;

    { T<grammar.recognizerName> }

```

```

constructor T<grammar.recognizerName>.Create;
begin
  InitializeCyclicDFAs;
end;

constructor T<grammar.recognizerName>.Create(const AInput: ICharStream<grammar.delegators:{g}; const
A<g:delegateName(): IBaseRecognizer{<g.recognizerName>} >>);
begin
  Create(AInput, nil<grammar.delegators:{g}, A<g:delegateName()>>);
end;

constructor T<grammar.recognizerName>.Create(const AInput: ICharStream; const AState:
IRecognizerSharedState<grammar.delegators:{g}; const A<g:delegateName():
IBaseRecognizer{<g.recognizerName>} >>);
begin
  inherited Create(AInput, AState);
  InitializeCyclicDFAs; { TODO: Necessary in Delphi??? Not removed yet. }
  <if(memoize)>
  <if(grammar.grammarIsRoot)>
    State.RuleMemoCount := <numRules>+1;<\n> <! index from 1..n !>
  <endif>
  <endif>
  <grammar.directDelegates:
    {g<g:delegateName()> := T<g.recognizerName>.Create(AInput, State<trunc(g.delegators):{p|,
<p:delegateName()>>, Self);}; separator="\n">
  <grammar.delegators:
    {g<g:delegateName()> := Pointer(A<g:delegateName()>);}; separator="\n">
  <last(grammar.delegators):{g|gParent := Pointer(A<g:delegateName()>);}>
  <actions.(actionScope).memberInitializations>
end;
<actions.(actionScope).memberImplementations>
function T<grammar.recognizerName>.GetGrammarFileName: String;
begin
  Result := '<fileName>';
end;

<if(filterMode)>
<filteringNextToken()>
<endif>

<rules; separator="\n\n">
<synpreds:{p | <lexerSynpred(p)>}>

procedure T<grammar.recognizerName>.InitializeCyclicDFAs;
begin
  <cyclicDFAs:{dfa | FDFFA<dfa.decisionNumber> :=
TDFFA<dfa.decisionNumber>.Create(Self<@debugAddition()>);}; separator="\n">
  <cyclicDFAs:{dfa | <if(dfa.specialStateSTs)>FDFFA<dfa.decisionNumber>.SpecialStateTransitionHandler :=

```

```
DFA<dfa.decisionNumber>_SpecialStateTransition;<endif>}; separator="\n">
end;
```

```
<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>
end.>>
```

```
lexerRuleDeclaration(rule) ::= <<
procedure m<rule.ruleName>(<rule.ruleDescriptor.parameterScope:parameterScope(scope=rule)>);<\n>
>>
```

```
/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 *
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */
```

```
filteringNextToken() ::= <<
function T<grammar.recognizerName>.NextToken: IToken;
var
  M: Integer;
begin
  while (True) do
  begin
    if (Input.LA(1) = Integer(cscEOF)) then
      Exit(TToken.EOF_TOKEN);

    State.Token := nil;
    State.Channel := TToken.DEFAULT_CHANNEL;
    State.TokenStartCharIndex := Input.Index;
    State.TokenStartCharPositionInLine := Input.CharPositionInLine;
    State.TokenStartLine := Input.Line;
    State.Text := "";
    try
      M := Input.Mark();
      State.Backtracking := 1; <! means we won't throw slow exception !>
      State.Failed := False;
      mTokens();
      State.Backtracking := 0;
    <!
      mTokens backtracks with synpred at backtracking==2
      and we set the synpredgate to allow actions at level 1.
    !>
    if (State.Failed) then
    begin
      Input.Rewind(M);
      Input.Consume; <! // advance one char and try again !>
    end
  end
```

```

else
begin
    Emit;
    Exit(State.Token);
end;
except
on RE: ERecognitionException do
begin
    // shouldn't happen in backtracking mode, but...
    ReportError(RE);
    Recover(RE);
end;
end;
end;
end;

function T<grammar.recognizerName>.AlreadyParsedRule(const Input: IIntStream;
const RuleIndex: Integer): Boolean;
begin
if (State.Backtracking > 1) then
    Result := inherited AlreadyParsedRule(Input, RuleIndex)
else
    Result := False;
end;

procedure T<grammar.recognizerName>.Memoize(const Input: IIntStream; const RuleIndex,
RuleStartIndex: Integer);
begin
if (State.Backtracking > 1) then
    inherited Memoize(Input, RuleIndex, RuleStartIndex);
end;

>>

filteringActionGate() ::= "(State.Backtracking = 1)"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass, filterMode,
    ASTLabelType="ANTLRInterface", labelType, members, rewriteElementType) ::= <<
type
<rules: {r | <genericParserRuleReturnType(rule=r, ruleDescriptor=r.ruleDescriptor)>>>
I<grammar.recognizerName> = interface(I<@superClassName><superClass><@end>)
    <rules: {r | <genericParserRuleInterface(rule=r, ruleDescriptor=r.ruleDescriptor)>>>
end;

T<grammar.recognizerName> = class(T<@superClassName><superClass><@end>,
I<grammar.recognizerName>)

```

```

<if(grammar.grammarIsRoot)>
public
  const
    TOKEN_NAMES: array [0..<length(tokenNames)>+3] of String = (
      '\<invalid>',
      '\<EOR>',
      '\<DOWN>',
      '\<UP>',
      <tokenNames; separator=",\n">);<\n>
<endif>
public
  const
    <tokens:{<it.name> = <it.type>;}; separator="\n">
public
  // delegates
  <grammar.delegates: {g|<g.delegateName(): I<superClass>; {<g.recognizerName>}}; separator="\n">
public
  // delegators
  <grammar.delegators: {g|<g.delegateName(): Pointer; {<g.recognizerName>}}; separator="\n">
  <last(grammar.delegators):{g|gParent: Pointer; {<g.recognizerName>}}>

  <scopes:{<if(it.isDynamicGlobalScope)><globalAttributeScopeDeclaration(scope=it)><endif>}>
<@members>
  <! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
public
  constructor Create(const AInput: <inputStreamType><grammar.delegators:{g}; const A<g.delegateName():
IBaseRecognizer{<g.recognizerName>}>); overload;
  constructor Create(const AInput: <inputStreamType>; const AState:
IRecognizerSharedState<grammar.delegators:{g}; const A<g.delegateName():
IBaseRecognizer{<g.recognizerName>}>); overload;
<@end>
protected
  { IBaseRecognizer }
  function GetTokenNames: TStringArray; override;
  function GetGrammarFileName: String; override;
strict private
  <actions.(actionScope).memberDeclarations>
  <rules: {r | <genericParserRuleDeclaration(rule=r, ruleDescriptor=r.ruleDescriptor)>}>

  <! generate rule/method definitions for imported rules so they
  appear to be defined in this recognizer. !>
  // Delegated rules
  <grammar.delegatedRules:{ruleDescriptor| <delegatedRuleDeclaration(ruleDescriptor)>}>

  <synpreds:{p | <synpredDeclaration(p)>}; separator="\n">
  <cyclicDFAs:cyclicDFADeclaration()>
strict private
  FException: ERecognitionException;

```

```

    FLA: array [0..<grammar.numberOfDecisions>, 0..255] of Integer;
    FCnt: array [0..<grammar.numberOfDecisions>] of Byte;
    procedure InitializeCyclicDFAs;
<if(bitsets)>
public
    class var
        <bitsets.bitsetDecl(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>})>
public
    class procedure InitializeBitsets; static;<\n>
<endif>
end;

implementation

uses
    <grammar.delegates: {g|<g.recognizerName>,}; separator="\n">
    <grammar.delegates: {g|<g.recognizerName>,}; separator="\n">
    <actions.(actionScope).usesImplementation>
    SysUtils,
    StrUtils,
    Math;

{ T<grammar.recognizerName> }

constructor T<grammar.recognizerName>.Create(const AInput: <inputStreamType><grammar.delegates:{g|; const
A<g:delegateName(): IBaseRecognizer{<g.recognizerName>} }>);
begin
    Create(AInput, TRecognizerSharedState.Create<grammar.delegates: {g|, A<g:delegateName()>}>);
end;

constructor T<grammar.recognizerName>.Create(const AInput: <inputStreamType>;
const AState: IRecognizerSharedState<grammar.delegates:{g|; const A<g:delegateName()>};
IBaseRecognizer{<g.recognizerName>} }>);
begin
    inherited Create(AInput, AState);
    <@membersConstructor>
    <@end>
    <parserCtorBody()>
    <grammar.directDelegates: {g|<g:delegateName()> := T<g.recognizerName>.Create(Input,
State<trunc(g.delegates):{p|, <p:delegateName()>}>, Self);}; separator="\n">
    <grammar.indirectDelegates: {g | <g:delegateName()> := <g.delegate:delegateName()>.<g:delegateName()>;};
separator="\n">
    <last(grammar.delegates): {g|gParent := Pointer(A<g:delegateName()>)}>
    <rules: {r | <ruleAttributeScopeInit(scope=r.ruleDescriptor.ruleScope)>}>
    <scopes: {<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
    <actions.(actionScope).memberInitializations>
end;
    <actions.(actionScope).memberImplementations>

```



```

<grammar.delegatedRules:{ruleDescriptor| <delegatedRuleImplementation(ruleDescriptor)>}; separator="\n">
procedure T<grammar.recognizerName>.InitializeCyclicDFAs;
begin
  <cyclicDFAs:{ dfa | FDFA<dfa.decisionNumber> := TDFA<dfa.decisionNumber>.Create(Self);}; separator="\n">
  <cyclicDFAs:{ dfa | <if(dfa.specialStateSTs)>FDFA<dfa.decisionNumber>.SpecialStateTransitionHandler :=
DFA<dfa.decisionNumber>_SpecialStateTransition;<endif>}; separator="\n">
end;

<if(bitsets)>
class procedure T<grammar.recognizerName>.InitializeBitsets;
begin
  <bitsets:bitset(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>}, words64=it.bits)>
end;
<endif>

<@membersImplementation>
<@end>

function T<grammar.recognizerName>.GetTokenNames: TStringArray;
var
  I: Integer;
begin
  SetLength(Result,Length(T<grammar.composite.rootGrammar.recognizerName>.TOKEN_NAMES));
  for I := 0 to Length(T<grammar.composite.rootGrammar.recognizerName>.TOKEN_NAMES) - 1 do
    Result[I] := T<grammar.composite.rootGrammar.recognizerName>.TOKEN_NAMES[I];
  end;

function T<grammar.recognizerName>.GetGrammarFileName: String;
begin
  Result := '<fileName>';
end;

<rules; separator="\n\n">
<synpreds:{p | <synpred(p)>}>

<cyclicDFAs:cyclicDFA()> <!-- dump tables for all DFA !-->
<if(bitsets)>
initialization
  T<grammar.recognizerName>.InitializeBitsets;<\n>
<endif>
end.>>

delegatedRuleDeclaration(ruleDescriptor) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
function <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>):
I<returnType()>;<\n>
<else>

```

```

<if(ruleDescriptor.hasSingleReturnValue)>
function <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>):
<returnType()>;<\n>
<else>
procedure <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>);<\n>
<endif>
<endif>
>>

delegatedRuleImplementation(ruleDescriptor) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
function
T<grammar.recognizerName>.<ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>
): I<returnType()>;<\n>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
function
T<grammar.recognizerName>.<ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>
): <returnType()>;<\n>
<else>
procedure
T<grammar.recognizerName>.<ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>
);<\n>
<endif>
<endif>
begin
<if(ruleDescriptor.hasReturnValue)>Result :=<endif>
T<ruleDescriptor.grammar.recognizerName>(<ruleDescriptor.grammar:delegateName()>.Implementor).<ruleDescriptor.name>(<ruleDescriptor.parameterScope.attributes: { a|<a.name> }; separator=" , ">);
end;

>>

parserCtorBody() ::= <<
InitializeCyclicDFAs;
<if(memoize)>
<if(grammar.grammarIsRoot)>
State.RuleMemoCount := <length(grammar.allImportedRules)>+1;<\n> <! index from 1..n !>
<endif>
<endif>
<grammar.delegators: { g|<g:delegateName()> := Pointer(A<g:delegateName()>); separator="\n">
>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets, ASTLabelType, superClass="Parser",
labelType="Token", members={<actions.parser.members>}) ::= <<
<genericParser(inputStreamType="ITokenStream", rewriteElementType="Token", ...)>
>>

```

```

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules, numRules, bitsets,
labelType={ <ASTLabelType> }, ASTLabelType="object", superClass="TreeParser",
members={ <actions.treeparser.members> }, filterMode) ::= <<
<genericParser(inputStreamType="ITreeNodeStream", rewriteElementType="Node", ...)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start "<ruleName>"
procedure
T<grammar.recognizerName>.<ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope(scope=it)>)
;
var
Alt: array [0..<grammar.numberOfDecisions>] of Integer;
<ruleLabelDefVars()>
begin
<ruleLabelDefs()>
<if(trace)>
TraceIn('<ruleName>_fragment', <ruleDescriptor.index>);
try
<block>
finally
TraceOut('<ruleName>_fragment', <ruleDescriptor.index>);
end;
<else>
<block>
<endif>
end;
// $ANTLR end "<ruleName>"
>>

synpredDecls(name) ::= <<
SynPredPointer <name>;<\n>
>>

synpred(name) ::= <<

function T<grammar.recognizerName>.<name>: Boolean;

```

```

var
  Start: Integer;
  Success: Boolean;
begin
  State.Backtracking := State.Backtracking + 1;
  <@start(>
  Start := Input.Mark;
  try
    <name>_fragment(); // can never throw exception
  except
    on RE: ERecognitionException do
      WriteLn('Impossible: ' + RE.ToString);
  end;
  Success := not State.Failed;
  Input.Rewind(Start);
  <@stop(>
  State.Backtracking := State.Backtracking - 1;
  State.Failed := False;
  Result := Success;
end;<\n>
>>

lexerSynpred(name) ::= <<
<synpred(name)>
>>

lexerSynpredDeclaration(name) ::= <<
function <name>: Boolean;
procedure <name>_fragment;
>>

synpredDeclaration(name) ::= <<
function <name>: Boolean;
procedure <name>_fragment;
>>

ruleMemoization(name) ::= <<
<if(memoize)>
if ((State.Backtracking > 0) and AlreadyParsedRule(Input, <ruleDescriptor.index>)) then
  Exit(<ruleReturnValue(>);
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)><\n>if (State.Failed) then Exit(<ruleReturnValue(>);<\n><endif>
>>

```

```

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>if (State.Backtracking > 0) then
begin
  State.Failed := True;
  Exit(<ruleReturnValue(>);
end;<endif>
>>

genericParserRuleDeclaration(rule, ruleDescriptor) ::= <<
<if(ruleDescriptor.isSynPred)>
<else>
<ruleAttributeScopeDeclaration(scope=ruleDescriptor.ruleScope)>
<returnScopeDeclaration(scope=ruleDescriptor.returnScope)>
public
<if(ruleDescriptor.hasMultipleReturnValues)>
  function <rule.ruleName>: I<returnType(>;<\n>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
  function <rule.ruleName>: <returnType(>;<\n>
<else>
  procedure <rule.ruleName>;<\n>
<endif>
<endif>
<endif>
>>

genericParserRuleInterface(rule, ruleDescriptor) ::= <<
<if(ruleDescriptor.isSynPred)>
<else>
<if(ruleDescriptor.hasMultipleReturnValues)>
function <rule.ruleName>: I<returnType(>;<\n>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
function <rule.ruleName>: <returnType(>;<\n>
<else>
procedure <rule.ruleName>;<\n>
<endif>
<endif>
<endif>
>>

genericParserRuleReturnType(rule, ruleDescriptor) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(ruleDescriptor.isSynPred)>
<else>
I<returnType(> = interface(I<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope)
end;<\n>

```

```

<endif>
<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(scope=ruleDescriptor.returnScope)>

// $ANTLR start "<ruleName>"
(* <fileName>:<description> *)
<if(ruleDescriptor.hasMultipleReturnValues)>
function T<grammar.recognizerName>.<ruleName>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>):
I<returnType()>;
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
function T<grammar.recognizerName>.<ruleName>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>):
<returnType()>;
<else>
procedure
T<grammar.recognizerName>.<ruleName>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>);
<endif>
<endif>

var
<ruleDescriptor.actions.vars>
Locals: TLocalStorage;
<if(ruleDescriptor.hasMultipleReturnValues)>
RetVal: I<returnType()>;<\n>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
RetVal: <returnType()>;<\n>
<else>
<endif>
<endif>
Alt: array [0..<grammar.numberOfDecisions>] of Integer;
<ruleDeclarationVars()>
<ruleLabelDefVars()>
begin
Locals.Initialize;
try
<if(trace)>TraceIn('<ruleName>', <ruleDescriptor.index>);<endif>
<ruleScopeSetUp()>
<ruleDeclarations()>
<ruleLabelDefs()>
<ruleDescriptor.actions.init>

```

```

<@preamble()>
try
  try
    <ruleMemoization(name=ruleName)>
    <block>
    <ruleCleanUp()>
    <(ruleDescriptor.actions.after):execAction()>
<if(exceptions)>
  <exceptions: { e|<catch(decl=e.decl,action=e.action)><\n}> }>
<else>
<if(!emptyRule)>
<if(actions.(actionScope).rulecatch)>
  <actions.(actionScope).rulecatch>
<else>
  except
    on RE: ERecognitionException do
    begin
      ReportError(RE);
      Recover(Input,RE);
      <@setErrorReturnValue()>
    end;<\n>
<endif>
<endif>
<endif>
  end;
  finally
    <if(trace)>TraceOut("<ruleName>", <ruleDescriptor.index>);<endif>
    <memoize()>
    <ruleScopeCleanUp()>
    <finally>
  end;
  <@postamble()>
  finally
    Locals.Finalize;
  end;
  Exit(<ruleReturnValue()>);
end;
// $ANTLR end "<ruleName>"
>>

catch(decl,action) ::= <<
catch (<e.decl>)
{
  <e.action>
}
>>

ruleDeclarations() ::= <<

```

```

<if(ruleDescriptor.hasMultipleReturnValues)>
RetVal := T<returnType().Create;
RetVal.Start := Input.LT(1);<\n>
<else>
<ruleDescriptor.returnScope.attributes: { a |
<a.name> := <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
<ruleDescriptor.name>_StartIndex := Input.Index();
<endif>
>>

ruleDeclarationVars() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<else>
<ruleDescriptor.returnScope.attributes: { a |
<a.name>: <a.type>;
}>
<endif>
<if(memoize)>
<ruleDescriptor.name>_StartIndex: Integer;
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: {<it>Stack.Push(T<it>Scope.Create);}; separator="\n">
<ruleDescriptor.ruleScope: {<it.name>Stack.Push(T<it.name>Scope.Create);}; separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: {<it>Stack.Pop();}; separator="\n">
<ruleDescriptor.ruleScope: {<it.name>Stack.Pop();}; separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels]: {<it.label.text> := nil;}; separator="\n">
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]: {list_<it.label.text> := nil;}; separator="\n">
<ruleDescriptor.ruleLabels:ruleLabelDef(label=it); separator="\n">
<ruleDescriptor.ruleListLabels: {ll<ll.label.text> := nil;}; separator="\n">
>>

ruleLabelDefVars() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels]: {<it.label.text>: I<labelType>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]: {list_<it.label.text>: IList<IANTLRInterface>;};
separator="\n">
<ruleDescriptor.ruleLabels:ruleLabelDefVar(label=it); separator="\n">
<ruleDescriptor.ruleListLabels: {ll<ll.label.text>: <ruleLabelType(referencedRule=ll.referencedRule)>;};

```



```
separator="\n">
```

```
>>
```

```
lexerRuleLabelDefs() ::= <<
```

```
<[ruleDescriptor.tokenLabels,
```

```
ruleDescriptor.tokenListLabels,
```

```
ruleDescriptor.ruleLabels]
```

```
:{<it.label.text> := nil;}; separator="\n"
```

```
>
```

```
<ruleDescriptor.charLabels:{int <it.label.text>;}; separator="\n">
```

```
<[ruleDescriptor.tokenListLabels,
```

```
ruleDescriptor.ruleListLabels,
```

```
ruleDescriptor.ruleListLabels]
```

```
:{List_<it.label.text> := nil;}; separator="\n"
```

```
>
```

```
>>
```

```
lexerRuleLabelDefDeclarations() ::= <<
```

```
<[ruleDescriptor.tokenLabels,
```

```
ruleDescriptor.tokenListLabels,
```

```
ruleDescriptor.ruleLabels]
```

```
:{<it.label.text>: I<labelType>;}; separator="\n"
```

```
>
```

```
<ruleDescriptor.charLabels:{int <it.label.text>;}; separator="\n">
```

```
<[ruleDescriptor.tokenListLabels,
```

```
ruleDescriptor.ruleListLabels,
```

```
ruleDescriptor.ruleListLabels]
```

```
:{List_<it.label.text>: IList;}; separator="\n"
```

```
>
```

```
>>
```

```
ruleReturnValue() ::= <<
```

```
<if(!ruleDescriptor.isSynPred)>
```

```
<if(ruleDescriptor.hasReturnValue)>
```

```
<if(ruleDescriptor.hasSingleReturnValue)>
```

```
<ruleDescriptor.singleValueReturnName>
```

```
<else>
```

```
RetVal
```

```
<endif>
```

```
<else>
```

```
<! nil !>
```

```
<endif>
```

```
<endif>
```

```
>>
```

```
ruleCleanup() ::= <<
```

```
<if(ruleDescriptor.hasMultipleReturnValues)>
```

```
<if(!TREE_PARSER)>
```

```

RetVal.Stop := Input.LT(-1);
<endif>
<endif>
>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if (State.Backtracking > 0) then
Memoize(Input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex);
<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
// $ANTLR start "<ruleName>"
<ruleDescriptor.parameterScope>
procedure
T<grammar.recognizerName>.m<ruleName>(<ruleDescriptor.parameterScope:parameterScope(scope=it)>);
var
<ruleDescriptor.actions.vars>
Locals: TLocalStorage;
TokenType, Channel: Integer;
Alt: array [0..<grammar.numberofDecisions>] of Integer;
<lexerRuleLabelDefDeclarations()>
begin
Locals.Initialize;
try
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<if(trace)>TraceIn("<ruleName>", <ruleDescriptor.index>);<endif>
<ruleScopeSetUp()>
<ruleDeclarations()>
try
<if(nakedBlock)>
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs()>
<ruleDescriptor.actions.init>
<block><\n>
<else>
TokenType := <ruleName>;
Channel := DEFAULT_TOKEN_CHANNEL;
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs()>
<ruleDescriptor.actions.init>
<block>

```

```

    <ruleCleanUp()>
    State.TokenType := TokenType;
    State.Channel := Channel;
    <(ruleDescriptor.actions.after):execAction()>
<endif>
    finally
    <if(trace)>TraceOut("<ruleName>", <ruleDescriptor.index>);<endif>
    <ruleScopeCleanUp()>
    <memoize()>
    end;
    finally
    Locals.Finalize;
    end;
end;
// $ANTLR end "<ruleName>"
>>

```

```

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */

```

```

*/
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
procedure T<grammar.recognizerName>.mTokens;
var
    Alt: array [0..

```

```

// S U B R U L E S

```

```

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
(* <fileName>:<description> *)
Alt[<decisionNumber>] := <maxAlt>;
<decls>
<@predecision()>
<decision>
<@postdecision()>
<@prebranch()>
case Alt[<decisionNumber>] of
    <alts:altSwitchCase()>
end;

```

```

<@postbranch(>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
(* <fileName>:<description> *)
Alt[<decisionNumber>] := <maxAlt>;
<decls>
<@predecision(>
<decision>
<@postdecision(>
case Alt[<decisionNumber>] of
<alts:altSwitchCase(>
end;
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
(* <fileName>:<description> *)
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
(* <fileName>:<description> *)
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
(* <fileName>:<description> *)
FCnt[<decisionNumber>] := 0;
<decls>
<@preloop(>
while (True) do
begin
Alt[<decisionNumber>] := <maxAlt>;
<@predecision(>
<decision>
<@postdecision(>
case Alt[<decisionNumber>] of
<alts:altSwitchCase(>

```

```

else
begin
  if (FCnt[<decisionNumber>] >= 1) then
    Break;
    <ruleBacktrackFailure()>
    raise EEarlyExitException.Create(<decisionNumber>, Input);
    <@earlyExitException()>
  end;
end;
Inc(FCnt[<decisionNumber>]);
end;
<@postloop()>
>>

```

positiveClosureBlockSingleAlt ::= positiveClosureBlock

```

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
(* <fileName>:<description> *)
<decls>
<@preloop()>
while (True) do
begin
  Alt[<decisionNumber>] := <maxAlt>;
  <@predecision()>
  <decision>
  <@postdecision()>
  case Alt[<decisionNumber>] of
    <alts:altSwitchCase()>
  else
    Break;
  end;
end;
<@postloop()>
>>

```

closureBlockSingleAlt ::= closureBlock

```

/** Optional blocks (x)? are translated to (x|) by before code generation
* so we can just use the normal block template
*/
optionalBlock ::= block

```

optionalBlockSingleAlt ::= block

```

/** A case in a switch that jumps to an alternative given the alternative
* number. A DFA predicts the alternative and then a simple switch

```

```

* does the jump to the code that actually matches that alternative.
*/
altSwitchCase() ::= <<
<i>:
<@prealt()>
<it><\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
(* <fileName>:<description> *)
begin
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
end;
>>

/** What to emit when there is no rewrite. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element() ::= <<
<@prematch()>
<it.el>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(label)><label> := <endif>Match(Input, <token>,
FOLLOW_<token>_in_<ruleName><elementIndex>)<if(label)> as
I<labelType><endif>;<\n><checkRuleBacktrackFailure()>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

listLabel(label,elem) ::= <<
if (list_<label> = nil) then list_<label> := TList<IANTLRInterface>.Create;
list_<label>.Add(<elem>);<\n>

```

```
>>
```

```
/** match a character */  
charRef(char,label) ::= <<  
<if(label)>  
<label> := Input.LA(1);<\n>  
<endif>  
Match(<char>); <checkRuleBacktrackFailure()>  
>>
```

```
/** match a character range */  
charRangeRef(a,b,label) ::= <<  
<if(label)>  
<label> := Input.LA(1);<\n>  
<endif>  
MatchRange(<a>, <b>); <checkRuleBacktrackFailure()>  
>>
```

```
/** For now, sets are interval tests and must be tested inline */  
matchSet(s,label,elementIndex,postmatchCode="") ::= <<  
<if(label)>  
<if(LEXER)>  
<label> := Input.LA(1);<\n>  
<else>  
<label> := Input.LT(1) as I<labelType>;<\n>  
<endif>  
<endif>  
if (<s>) then  
begin  
Input.Consume;  
<postmatchCode>  
<if(!LEXER)>  
State.ErrorRecovery := False;<endif>  
<if(backtracking)>State.Failed := False;<endif>  
end  
else  
begin  
<ruleBacktrackFailure()>  
FException := EMismatchedSetException.Create(nil, Input);  
<@mismatchedSetException()>  
<if(LEXER)>  
Recover(FException);  
raise FException;<\n>  
<else>  
raise FException;  
<! use following code to make it recover inline; remove throw mse;  
RecoverFromMismatchedSet(input,mse,FOLLOW_set_in_<ruleName><elementIndex>);  
!>
```

```

<endif>
end;<\n>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <<
<if(label)>
Locals.AsInteger['<label>Start'] := CharIndex;
Match(<string>); <checkRuleBacktrackFailure()>
<label> := TCommonToken.Create(Input, TToken.INVALID_TOKEN_TYPE, TToken.DEFAULT_CHANNEL,
Locals.AsInteger['<label>Start'], CharIndex-1);
<else>
Match(<string>); <checkRuleBacktrackFailure()>
<endif>
>>

wildcard(label,elementIndex) ::= <<
<if(label)>
<label> := Input.LT(1) as I<labelType>;<\n>
<endif>
MatchAny(input); <checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(label,elementIndex) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> := Input.LA(1);<\n>
<endif>
MatchAny(); <checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabel(elem=label,...)>
>>

```



```

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
PushFollow(FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)>
<label> := <if(scope)><scope:delegateName().<endif><rule.name>(<args; separator=", ">);<\n>
<else>
<if(scope)>T<scope.recognizerName>(IANTLRObject(<scope:delegateName()>).Implementor).<endif><rule.name>
>(<args; separator=", ">);<\n>
<endif>
State.FollowingStackPointer := State.FollowingStackPointer - 1;
<checkRuleBacktrackFailure()>
>>

/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>

/** A lexer rule reference.
 *
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
Locals.AsInteger['<label>Start<elementIndex>'] := CharIndex;
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator=", ">);
<checkRuleBacktrackFailure()>
<label> := TCommonToken.Create(Input, TToken.INVALID_TOKEN_TYPE, TToken.DEFAULT_CHANNEL,
Locals.AsInteger['<label>Start<elementIndex>'], CharIndex - 1);
<else>
<if(scope)><scope:delegateName().Implementor as T<scope.recognizerName>.<endif>m<rule.name>(<args;
separator=", ">); <checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>
>>

```

```

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
Locals.AsInteger['<label>Start<elementIndex>'] := CharIndex;
Match(EOF); <checkRuleBacktrackFailure()>
Locals['<label>'] := TCommonToken.Create(Input, EOF, TToken.DEFAULT_CHANNEL,
Locals.AsInteger['<label>Start<elementIndex>'], CharIndex-1);
<else>
Match(EOF); <checkRuleBacktrackFailure()>
<endif>
>>

```

```

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if (Input.LA(1) = TToken.DOWN) then
begin
Match(Input, TToken.DOWN, nil); <checkRuleBacktrackFailure()>
<children:element()>
Match(Input, TToken.UP, nil); <checkRuleBacktrackFailure()>
end;
<else>
Match(Input, TToken.DOWN, nil); <checkRuleBacktrackFailure()>
<children:element()>
Match(Input, TToken.UP, nil);<\n><checkRuleBacktrackFailure()>
<endif>
>>

```

```

/** Every predicate is used as a validating predicate (even when it is
* also hoisted into a prediction expression).
*/
validateSemanticPredicate(pred,description) ::= <<
if (not (<evalPredicate(...)>)) then
begin
<ruleBacktrackFailure()>
raise EFailedPredicateException.Create(Input, '<ruleName>', '<description>');
end;<\n>
>>

```

```

// F i x e d D F A (if-then-else)

```

```

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
FLA[<decisionNumber>,<stateNumber>] := Input.LA(<k>);<\n>
<edges; separator="\nelse ">

```

```

else
begin
<if(eotPredictsAlt)>
  Alt[<decisionNumber>] := <eotPredictsAlt>;<\n>
<else>
  <ruleBacktrackFailure()>
  raise ENoViableAltException.Create('<description>', <decisionNumber>, <stateNumber>, Input);<\n>
<endif>
end;
>>

```

```

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
FLA[<decisionNumber>,<stateNumber>] := Input.LA(<k>);<\n>
<edges; separator="\nelse ">;
>>

```

```

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
FLA[<decisionNumber>,<stateNumber>] := Input.LA(<k>);
<edges; separator="\nelse ">;<\n>
<if(eotPredictsAlt)>
<if(!edges)>
Alt[<decisionNumber>] := <eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else
begin
  Alt[<decisionNumber>] := <eotPredictsAlt>;
end;<\n>
<endif>
<endif>
>>

```

```

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "Alt[<decisionNumber>] := <alt>;"

```

```

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.

```

```

*/
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ((<labelExpr>)<if(predicates)> and (<predicates>)<endif>) then
begin
  <targetState>
end <! no ; here !>
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
case Input.LA(<k>) of
  <edges; separator="\n">
else
begin
<if(eotPredictsAlt)>
  Alt[<decisionNumber>] := <eotPredictsAlt>;
<else>
  <ruleBacktrackFailure()>
  <@noViableAltException()>
  raise ENoViableAltException.Create('<description>', <decisionNumber>, <stateNumber>, Input);<\n>
<endif>
end;
end;<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
case Input.LA(<k>) of
  <edges; separator="\n">
end;<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
case Input.LA(<k>) of
  <edges; separator="\n"><\n>
<if(eotPredictsAlt)>
else
  Alt[<decisionNumber>] := <eotPredictsAlt>;<\n>
<endif>
end;<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels:{<it>}; separator=",\n">:
begin

```

```

    <targetState>
end;
>>

// C y c l i c   D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
Alt[<decisionNumber>] := FDFA<decisionNumber>.Predict(Input);
>>

/* Dump DFA tables.
 */
cyclicDFADeclaration(dfa) ::= <<
strict protected
type
  TDFA<dfa.decisionNumber> = class(TDFA)
protected
  { IDFA }
  function Description: String; override;
public
  constructor Create(const ARecognizer: IBaseRecognizer);
end;
var
  FDFA<dfa.decisionNumber>: IDFA;
<if(dfa.specialStateSTs)>
strict protected
function DFA<dfa.decisionNumber>_SpecialStateTransition(const DFA: IDFA; S: Integer;
  const AInput: IIntStream): Integer;<endif>
>>

cyclicDFA(dfa) ::= <<
{ T<grammar.recognizerName>.TDFA<dfa.decisionNumber> }

constructor T<grammar.recognizerName>.TDFA<dfa.decisionNumber>.Create(const ARecognizer:
IBaseRecognizer);
const
DFA<dfa.decisionNumber>_EOT = '<dfa.javaCompressedEOT; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_EOF = '<dfa.javaCompressedEOF; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_MIN = '<dfa.javaCompressedMin; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_MAX = '<dfa.javaCompressedMax; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_ACCEPT = '<dfa.javaCompressedAccept; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_SPECIAL = '<dfa.javaCompressedSpecial; wrap=""+\n  "'>;
DFA<dfa.decisionNumber>_TRANSITION: array [0..<length(dfa.javaCompressedTransition)>-1] of String = (
  <dfa.javaCompressedTransition: {s|<s; wrap=""+\n">'}; separator=",\n">);

```

```

begin
  inherited Create;
  Recognizer := ARecognizer;
  DecisionNumber := <dfa.decisionNumber>;
  EOT := TDFA.UnpackEncodedString(DFA<dfa.decisionNumber>_EOT);
  EOF := TDFA.UnpackEncodedString(DFA<dfa.decisionNumber>_EOF);
  Min := TDFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_MIN);
  Max := TDFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_MAX);
  Accept := TDFA.UnpackEncodedString(DFA<dfa.decisionNumber>_ACCEPT);
  Special := TDFA.UnpackEncodedString(DFA<dfa.decisionNumber>_SPECIAL);
  Transition := TDFA.UnpackEncodedStringArray(DFA<dfa.decisionNumber>_TRANSITION);
end;

function T<grammar.recognizerName>.TDFA<dfa.decisionNumber>.Description: String;
begin
  Result := '<dfa.description>';
end;<\n>
<if(dfa.specialStateSTs)>
function T<grammar.recognizerName>.DFA<dfa.decisionNumber>_SpecialStateTransition(const DFA: IDFA; S:
Integer;
const AInput: IIntStream): Integer;
var
  Locals: TLocalStorage;
  <if(LEXER)>
  Input: IIntStream;
  <endif>
  <if(PARSER)>
  Input: ITokenStream;
  <endif>
  <if(TREE_PARSER)>
  Input: ITreeNodeStream;
  <endif>
  _S: Integer;
  NVAE: ENoViableAltException;
begin
  Result := -1;
  Locals.Initialize;
  try
    <if(LEXER)>
    Input := AInput;
    <endif>
    <if(PARSER)>
    Input := AInput as ITokenStream;
    <endif>
    <if(TREE_PARSER)>
    Input := AInput as ITreeNodeStream;
    <endif>
    _S := S;

```

```

case S of
  <dfa.specialStateSTs:{state | <i0>: begin<! compressed special state numbers 0..n-1 !>
    <state> <\n> end;}; separator="\n">
  end;
<if(backtracking)>
  if (State.Backtracking > 0) then
  begin
    State.Failed := True;
    Exit(-1);
  end;<\n>
<endif>
  NVAE := ENoViableAltException.Create(DFA.Description, <dfa.decisionNumber>, _S, Input);
  DFA.Error(NVAE);
  raise NVAE;
finally
  Locals.Finalize;
end;
end;<\n>
<endif>
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
FLA[<decisionNumber>,<stateNumber>] := Input.LA(1);<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
Locals.AsInteger['index<decisionNumber>_<stateNumber>'] := Input.Index;
Input.Rewind;<\n>
<endif>
S := -1;
<edges; separator="\nelse ">;
<if(semPredState)> <! return input cursor to state before we rewound !>
Input.Seek(Locals.AsInteger['index<decisionNumber>_<stateNumber>']);<\n>
<endif>
if (S >= 0) then
  Exit(S);
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ((<labelExpr>)<if(predicates)> and (<predicates>)<endif>) then
  S := <targetStateNumber>
>>

/** An edge pointing at end-of-token; essentially matches any char;

```

```

* always jump to the target.
*/
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
S := <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "((<left>) and (<right>))"

orPredicates(operands) ::= "<operands:{o|(<o>)}; separator=\" or \">"

notPredicate(pred) ::= "!(<evalPredicate(...)>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "FLA[<decisionNumber>,<stateNumber>] = <atomAsInt>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
* somewhere. Must ask for the lookahead directly.
*/
isolatedLookaheadTest(atom,k,atomAsInt) ::= "Input.LA(<k>) = <atomAsInt>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <<
((FLA[<decisionNumber>,<stateNumber>] \>= <lowerAsInt>) and (FLA[<decisionNumber>,<stateNumber>] \<=
<upperAsInt>))
>>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(Input.LA(<k>) \>=
<lowerAsInt>) and (Input.LA(<k>) \<= <upperAsInt>)"

setTest(ranges) ::= "<ranges; separator=\" or (\">"

// A T T R I B U T E S

globalAttributeScope(scope) ::= <<
<scope.name>Stack := TStackList<I<scope.name>Scope>.Create;<\n>
<endif>
>>

globalAttributeScopeDeclaration(scope) ::= <<
<if(scope.attributes)>
strict protected
type
I<scope.name>Scope = interface(IANTLRObjct)

```



```

end;
T<scope.name>Scope = class(TANTLRObjct, I<scope.name>Scope)
protected
  <scope.attributes:{<it.name>: <it.type>;}; separator="\n">
end;
strict protected
<scope.name>Stack: IStackList<I<scope.name>Scope>;
<endif>
>>

```

```

ruleAttributeScopeDeclaration(scope) ::= <<
<if(scope.attributes)>
strict protected
type
  I<scope.name>Scope = interface(IANTLRObjct)
  end;
  T<scope.name>Scope = class(TANTLRObjct, I<scope.name>Scope)
  protected
    <scope.attributes:{<it.name>: <it.type>;}; separator="\n">
  end;
strict protected
  <scope.name>Stack: IStackList<I<scope.name>Scope>;
<endif>
>>

```

```

ruleAttributeScope(scope) ::= <<
<! protected Stack <scope.name>Stack = new Stack();<\n !>
>>

```

```

ruleAttributeScopeInit(scope) ::= <<
<if(scope)>
<scope.name>Stack := TStackList<I<scope.name>Scope>.Create;<\n>
<endif>
>>

```

```

returnStructName() ::= "<it.name>_return"

```

```

returnType() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor:returnStructName()>
<! I<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope !>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
<! Pointer/void !>
<endif>
<endif>

```

```

>>

/** Generate the C# type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
I<referencedRule.name>_return
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
<endif>
>>

delegateName() ::= <<
<if(it.label)><it.label><else>g<it.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
<csharpTypeInitMap.(typeName)>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
<label.label.text> := <initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;<\n>
>>

ruleLabelDefVar(label) ::= <<
<label.label.text>: <ruleLabelType(referencedRule=label.referencedRule)>;
>>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
{ T<ruleDescriptor:returnStructName()> }

<scope.attributes:{public <it.decl>;}; separator="\n">
<@ruleReturnMembers()>
<endif>

```

>>

```
returnScopeDeclaration(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
public
type
  T<ruleDescriptor:returnStructName()> =
class(T<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope, I<ruleDescriptor:returnStructName()>)
  <scope.attributes:{public <it.decl>;}; separator="\n">
  <@ruleReturnMembers()>
  end;
<endif>
>>
```

```
parameterScope(scope) ::= <<
<scope.attributes:{<it.decl>}; separator=", ">
>>
```

```
parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name> := <expr>;"
```

```
scopeAttributeRef(scope,attr,index,negIndex) ::= <<
<if(negIndex)>
(<scope>Stack[<scope>Stack.Count-<negIndex>-1] as T<scope>Scope).<attr.name>
<else>
<if(index)>
(<scope>Stack[<index>] as T<scope>Scope).<attr.name>
((<scope>_scope)<scope>_stack[<index>]).<attr.name>
<else>
(<scope>Stack.Peek.Implementor as T<scope>Scope).<attr.name>
<endif>
<endif>
>>
```

```
scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <<
<if(negIndex)>
(<scope>Stack[<scope>Stack.Count-<negIndex>-1] as T<scope>Scope).<attr.name> := <expr>;<\n>
<else>
<if(index)>
(<scope>Stack[<index>] as T<scope>Scope).<attr.name> := <expr>;<\n>
<else>
(<scope>Stack.Peek.Implementor as T<scope>Scope).<attr.name> := <expr>;<\n>
<endif>
<endif>
>>
```

/** \$x is either global scope or x is rule with dynamic scope; refers

* to stack itself not top of stack. This is useful for predicates

```

* like {$function.size()>0 && $function::name.equals("foo")}?
*/
isolatedDynamicScopeRef(scope) ::= "<scope>Stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
(IfThen(Assigned(<scope>),Def(<scope>).<attr.name>,<initValue(attr.type)>))
<else>
<scope>
<endif>
>>

returnAttributeRef(ruleDescriptor,attr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
RetVal.<attr.name>
<else>
<attr.name>
<endif>
>>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
RetVal.<attr.name> := <expr>;
<else>
<attr.name> := <expr>;
<endif>
>>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "(Def(<scope>).Text)"
tokenLabelPropertyRef_type(scope,attr) ::= "(Def(<scope>).TokenType)"
tokenLabelPropertyRef_line(scope,attr) ::= "(Def(<scope>).Line)"
tokenLabelPropertyRef_pos(scope,attr) ::= "(Def(<scope>).CharPositionInLine)"
tokenLabelPropertyRef_channel(scope,attr) ::= "(Def(<scope>).Channel)"
tokenLabelPropertyRef_index(scope,attr) ::= "(Def(<scope>).TokenIndex)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "(StrToIntDef(Def(<scope>).Text,0))"

ruleLabelPropertyRef_start(scope,attr) ::= "(IfThen(Assigned(<scope>), Def(<scope>).Start, nil) as I<labelType>)"

```

```

ruleLabelPropertyRef_stop(scope,attr) ::= "(Def(<scope>).Stop as I<labelType>)"
ruleLabelPropertyRef_tree(scope,attr) ::= "(Def(Def(<scope>).Tree as I<ASTLabelType>))"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
IfThen(Assigned(<scope>), Input.TokenStream.ToString(
  Input.TreeAdaptor.GetTokenStartIndex(Def(<scope>).Start),
  Input.TreeAdaptor.GetTokenStopIndex(Def(<scope>).Start)), ")
<else>
IfThen(Assigned(<scope>), Input.ToString(
  (Def(<scope>).Start) as IToken,(Def(<scope>).Stop) as IToken), ")
<endif>
>>
ruleLabelPropertyRef_st(scope,attr) ::= "((<scope> != null) ? <scope>.ST : null)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "(Def(<scope>).TokenType)"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "(Def(<scope>).Line)"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "(IfThen(Assigned(<scope>),Def(<scope>).CharPositionInLine,-1))"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "(Def(<scope>).Channel)"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "(Def(<scope>).TokenIndex)"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "(Def(<scope>).Text)"
lexerRuleLabelPropertyRef_int(scope,attr) ::= "(StrToIntDef(Def(<scope>).Text,0))"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "(RetVal.Start as I<labelType>)"
rulePropertyRef_stop(scope,attr) ::= "(RetVal.Stop as I<labelType>)"
rulePropertyRef_tree(scope,attr) ::= "(RetVal.Tree as I<ASTLabelType>)"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
Input.TokenStream.ToString(
  Input.TreeAdaptor.GetTokenStartIndex(RetVal.Start),
  Input.TreeAdaptor.GetTokenStopIndex(RetVal.Start))
<else>
Input.ToString(RetVal.Start as IToken,Input.LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "RetVal.ST"

lexerRulePropertyRef_text(scope,attr) ::= "Text"
lexerRulePropertyRef_type(scope,attr) ::= "TokenType"
lexerRulePropertyRef_line(scope,attr) ::= "State.TokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "State.TokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "Channel"
lexerRulePropertyRef_start(scope,attr) ::= "State.TokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(CharIndex-1)"

```

```

lexerRulePropertyRef_int(scope,attr) ::= "StrToInt(<scope>.Text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "RetVal.Tree := <expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "RetVal.ST := <expr>;"

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>
<if(actions.(actionScope).synpredgate)>
if (<actions.(actionScope).synpredgate>) then
begin
<action>
end;
<else>
if (State.Backtracking = 0) then
begin
<action>
end;<\n>
<endif>
<else>
<action>
<endif>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
<name> := TBitSet.Create([<words64: {<it>} ;separator=",">]);<\n>
>>

bitsetDecl(name) ::= <<
<name>: IBitSet;<\n>
>>

codeFileExtension() ::= ".pas"

true() ::= "True"
false() ::= "False"

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-

```

jar/org/antlr/codegen/templates/Delphi/Delphi.stg

No license file was found, but licenses were detected in source scan.

```
/*
 * [The "BSD license"]
 * Copyright (c) 2005-2008 Terence Parr
 * All rights reserved.
 *
 * Conversion to C#:
 * Copyright (c) 2008-2009 Sam Harwell, Pixel Mine, Inc.
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

/** Template overrides to add debugging to AST stuff. Dynamic inheritance
 * hierarchy is set up as ASTDbg : AST : Dbg : Java by code generator.
 */

parserMembers() ::= <<
// Implement this function in your helper file to use a custom tree adaptor
partial void InitializeTreeAdaptor();
protected DebugTreeAdaptor adaptor;

public ITreeAdaptor TreeAdaptor
{
get
{
```

```

    return adaptor;
  }
  set
  {
<if(grammar.grammarIsRoot)>
    this.adaptor = new DebugTreeAdaptor(dbg,adaptor);
<else>
    this.adaptor = (DebugTreeAdaptor)adaptor; // delegator sends dbg adaptor
<endif><\n>
    <grammar.directDelegates:{g|<g.delegateName()>.TreeAdaptor = this.adaptor;}>
  }
  }<\n>
>>

```

```

parserCtorBody() ::= <<
<super.parserCtorBody()>
>>

```

```

createListenerAndHandshake() ::= <<
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port,
<if(TREE_PARSER)>input.TreeAdaptor<else>adaptor<endif> );
DebugListener = proxy;
<inputStreamType> = new Debug<inputStreamType>( input, proxy );
try
{
    proxy.Handshake();
}
catch ( IOException ioe )
{
    ReportError( ioe );
}
>>

```

```

@ctorForRootGrammar.finally() ::= <<
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;
proxy.TreeAdaptor = adap;
>>

```

```

@ctorForProfilingRootGrammar.finally() ::= <<
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;
>>

```

```

@ctorForPredefinedListener.superClassRef() ::= ": base( input, dbg )"

```

```

@ctorForPredefinedListener.finally() ::= <<
<if(grammar.grammarIsRoot)><! don't create new adaptor for delegates !>

```



```
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;<\n>
<endif>
>>

//@rewriteElement.pregen() ::= "dbg.Location( <e.line>, <e.pos> );"
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/ASTDbg.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
 * [The "BSD license"]
 * Copyright (c) 2007-2008 Johannes Luber
 * Copyright (c) 2005-2007 Kunle Odutola
 * Copyright (c) 2011 Sam Harwell
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */

/** Templates for building ASTs during normal parsing.
 *
 * Deal with many combinations. Dimensions are:
 * Auto build or rewrite
 * no label, label, list label (label/no-label handled together)
```

```

* child, root
* token, set, rule, wildcard
*
* The situation is not too bad as rewrite (->) usage makes ^ and !
* invalid. There is no huge explosion of combinations.
*/

@rule.setErrorReturnValue() ::= <<
retval.Tree = (<ASTLabelType>)adaptor.ErrorNode(input, retval.Start, input.LT(-1), re);
<! System.out.WriteLine("<ruleName> returns "+((CommonTree)retval.tree).toStringTree()); !>
>>

// TOKEN AST STUFF

/** ID and output=AST */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<super.tokenRef(...)>
<if(backtracking)>if (state.backtracking == 0) {<endif>
<label>_tree = <createNodeFromToken(...)>;
adaptor.AddChild(root_0, <label>_tree);
<if(backtracking)>}<endif>
>>

/** ID! and output=AST (same as plain tokenRef) */
tokenRefBang(token,label,elementIndex) ::= "<super.tokenRef(...)>"

/** ID^ and output=AST */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={}) ::= <<
<super.tokenRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = <createNodeFromToken(...)>;
root_0 = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_0);
<if(backtracking)>}<endif>
>>

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */

```

```

tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRoot(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= <<
<super.matchSet(postmatchCode={ <if(backtracking)>if (<actions.(actionScope).synpredgate>)
<endif>adaptor.AddChild(root_0, <createNodeFromToken(...)>);}, ...)>
>>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={}) ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= "<super.matchSet(...)>"

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={}) ::= <<
<if(label)>
<label>=(<labelType>)input.LT(1);
<endif>
<super.matchSet(postmatchCode={ <if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>root_0 =
(<ASTLabelType>)adaptor.BecomeRoot(<createNodeFromToken(...)>, root_0);}, ...)>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>adaptor.AddChild(root_0, <label>.Tree);
>>

```

```

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>root_0 =
(<ASTLabelType>)adaptor.BecomeRoot(<label>.Tree, root_0);
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

// WILDCARD AST

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.Create(<label>);
adaptor.AddChild(root_0, <label>_tree);
<if(backtracking)>}<endif>
>>

wildcardBang(label,elementIndex) ::= "<super.wildcard(token=[],...)>"

wildcardRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.Create(<label>);
root_0 = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_0);
<if(backtracking)>}<endif>
>>

```

```

createNodeFromToken(label,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
new
<terminalOptions.node><(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(if(terminalOptions.text)>,<terminalOptions.text; format="string"><endif>)
<else>
(<ASTLabelType>)adaptor.Create(<(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(if(terminalOptions.text)>,<terminalOptions.text; format="string"><endif>)
<endif>
%>

```

```

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
retval.Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
adaptor.SetTokenBoundaries(retval.Tree, retval.Start, retval.Stop);
<if(backtracking)>}<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp2/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

[The "BSD license"]

Copyright (c) 2005-2006 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during normal parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* The situation is not too bad as rewrite (->) usage makes ^ and !

* invalid. There is no huge explosion of combinations.

*/

```
finishedBacktracking(block) ::= <<
```

```
<if(!ruleDescriptor.isSynPred)>
```

```
<if(backtracking)>
```

```
if <actions.(actionScope).synpredgate>:
```

```
  <block>
```

```
<else>
```

```
<block>
```

```
<endif>
```

```
<endif>
```

```
>>
```

```
@ruleBody.setErrorReturnValue() ::= <<
```

```
retval.tree = self._adaptor.errorNode(self.input, retval.start, self.input.LT(-1), re)
```

```
>>
```

```
// TOKEN AST STUFF
```

```
/** ID and output=AST */
```

```
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
```

```
<super.tokenRef(...)>
```

```
<finishedBacktracking({
```

```
<label>_tree = <createNodeFromToken(...)>
```

```
self._adaptor.addChild(root_0, <label>_tree
```

```
})>
```

```
>>
```

```
/** ID! and output=AST (same as plain tokenRef) */
```

```
tokenRefBang(token,label,elementIndex,terminalOptions={}) ::= "<super.tokenRef(...)>"
```

```
/** ID^ and output=AST */
```

```
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={}) ::= <<
```

```
<super.tokenRef(...)>
```

```

<finishedBacktracking({
<label>_tree = <createNodeFromToken(...)>
root_0 = self._adaptor.becomeRoot(<label>_tree, root_0)
})>
>>

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<listLabel(elem=label,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= <%
<super.matchSet(postmatchCode={<finishedBacktracking({self._adaptor.addChild(root_0,
<createNodeFromToken(...)>})>}, ...)>
%>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={}) ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= "<super.matchSet(...)>"

```

```

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={}) ::= <<
<if(label)>
<label> = self.input.LT(1)<\n>
<endif>
<super.matchSet(postmatchCode={ <finishedBacktracking({root_0 =
self._adaptor.becomeRoot(<createNodeFromToken(...)>, root_0)}>}, ...)>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<finishedBacktracking({self._adaptor.addChild(root_0, <label>.tree})>)>
>>

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<finishedBacktracking({root_0 = self._adaptor.becomeRoot(<label>.tree, root_0)}>)>
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.tree})>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabel(label, {<label>.tree})>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(label, {<label>.tree})>
>>

// WILDCARD AST

```



```
wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<finishedBacktracking({
<label>_tree = self._adaptor.createWithPayload(<label>)
self._adaptor.addChild(root_0, <label>_tree)
})>
>>
```

```
wildcardBang(label,elementIndex) ::= "<super.wildcard(...)>"
```

```
wildcardRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<finishedBacktracking({
<label>_tree = self._adaptor.createWithPayload(<label>)
root_0 = self._adaptor.becomeRoot(<label>_tree, root_0)
})>
>>
```

```
createNodeFromToken(label,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
<terminalOptions.node>(<label>) <! new MethodNode(IDLabel) !>
<else>
self._adaptor.createWithPayload(<label>)
<endif>
%>
```

```
ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<finishedBacktracking({
retval.tree = self._adaptor.rulePostProcessing(root_0)
self._adaptor.setTokenBoundaries(retval.tree, retval.start, retval.stop)
})>
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2008 Erik van Bilzen
```

```
Copyright (c) 2007-2008 Johannes Luber
```

```
Copyright (c) 2005-2007 Kunle Odutola
```

```
Copyright (c) 2005 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without

modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

group AST;

```
@outputFile.imports() ::= <<
<@super.imports()><if(!TREE_PARSER)><! tree parser would already have imported !>
Antlr.Runtime.Tree,<\n><endif>
>>
```

```
@genericParser.members() ::= <<
<@super.members()>
<parserMembers()>
>>
```

```
@genericParser.membersConstructor() ::= <<
<@super.membersConstructor()>
<parserMembersConstructor()>
>>
```

```
@genericParser.membersImplementation() ::= <<
<@super.membersImplementation()>
<parserMembersImplementation()>
>>
```

/** Add an adaptor property that knows how to build trees */

```
parserMembers() ::= <<
strict protected
    FAdaptor: ITreeAdaptor;
    procedure SetAdaptor(const Value: ITreeAdaptor);
```

```

    property Adaptor: ITreeAdaptor read FAdaptor;
public
    property TreeAdaptor: ITreeAdaptor read FAdaptor write SetAdaptor;

>>

parserMembersConstructor() ::= <<
FAdaptor := TCommonTreeAdaptor.Create;
>>

parserMembersImplementation() ::= <<
procedure T<grammar.recognizerName>.SetAdaptor(const Value: ITreeAdaptor);
begin
    FAdaptor := Value;
    <grammar.directDelegates:{g|<g:delegateName()>.TreeAdaptor := FAdaptor;}>
end;
>>

@returnScope.ruleReturnMembers() ::= <<
function T<grammar.recognizerName>.T<ruleDescriptor:returnStructName()>.GetTree: IANTLRInterface;
begin
    Result := FTree;
end;

procedure T<grammar.recognizerName>.T<ruleDescriptor:returnStructName()>.SetTree(const Value:
IANTLRInterface);
begin
    FTree := Value as I<ASTLabelType>;
end;
>>

@returnScopeDeclaration.ruleReturnMembers() ::= <<
strict private
    FTree: I<ASTLabelType>;
protected
    { IRuleReturnScope }
    function GetTree: IANTLRInterface; override;
    procedure SetTree(const Value: IANTLRInterface); override;
>>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
Root[0] := nil;<\n>
>>

ruleDeclarationVars() ::= <<
<super.ruleDeclarationVars()>

```

Root: array [0..63] of I<ASTLabelType>;

>>

```
ruleLabelDefs() ::= <<
<super.ruleLabelDefs()>
<ruleDescriptor.tokenLabels:{<it.label.text>_tree := nil;}; separator="\n">
<ruleDescriptor.tokenListLabels:{<it.label.text>_tree := nil;}; separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites:{Locals['Stream_<it>'] :=
TRewriteRule<rewriteElementType>Stream.Create(Adaptor,'token <it>');}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites:{Locals['Stream_<it>'] :=
TRewriteRuleSubtreeStream.Create(Adaptor,'rule <it>');}; separator="\n">
>>
```

```
ruleLabelDefVars() ::= <<
<super.ruleLabelDefVars()>
<ruleDescriptor.tokenLabels:{<it.label.text>_tree: I<ASTLabelType>;}; separator="\n">
<ruleDescriptor.tokenListLabels:{<it.label.text>_tree: I<ASTLabelType>;}; separator="\n">
>>
```

```
/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
```

```
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
Root[0] := Adaptor.GetNilNode as I<ASTLabelType>;
<endif>
<endif>
<endif>
>>
```

```
// Tracking Rule Elements
```

```
/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if (State.Backtracking = 0) then <endif>(Locals['Stream_<token>'] as
IRewriteRuleElementStream).Add(<label>);<\n>
>>
```

```
/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
 * to the tracking list stream_ID for use in the rewrite.
 */
```

```
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label,...)>
```

```

>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>(Locals['Stream_<token>'] as
IRewriteRuleElementStream).Add(<label>);<\n>
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label,...)>
>>

wildcardTrack(label,elementIndex) ::= <<
<super.wildcard(...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>(Locals['Stream_<rule.name>'] as
IRewriteRuleElementStream).Add(<label>.Tree);<\n>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabel(elem=label+".Tree",...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>(Locals['Stream_<rule>'] as
IRewriteRuleElementStream).Add(<label>.Tree);
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabel(elem=label+".Tree",...)>
>>

// R e w r i t e

rewriteCode(

```

```

alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
  referencedWildcardLabels,
  referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::=
<<

// AST REWRITE
// elements:    <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels:  <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
<if(backtracking)>
if (State.Backtracking = 0) then
begin<\n>
<endif>
<prevRuleRootRef(>).Tree := Root[0];
<rewriteCodeLabels(>
Root[0] := Adaptor.GetNilNode as I<ASTLabelType>;
<alts:rewriteAlt(> separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER)>
<if(rewriteMode)>
<prevRuleRootRef(>).Tree = (<ASTLabelType>)adaptor.rulePostProcessing(root[0]);
input.ReplaceChildren(adaptor.GetParent(retval.Start),
                    adaptor.GetChildIndex(retval.Start),
                    adaptor.GetChildIndex(_last),
                    retval.Tree);

<endif>
<endif>
<! if parser or rewrite!=true, we need to set result !>
<if(!TREE_PARSER)>
<prevRuleRootRef(>).Tree := Root[0];<\n>
<endif>
<if(!rewriteMode)>
<prevRuleRootRef(>).Tree := Root[0];<\n>
<endif>
<if(backtracking)>
end;
<endif>
>>

rewriteCodeLabels() ::= <<

```

```

<referencedTokenLabels
  :{Locals['Stream_<it>'] := TRewriteRule<rewriteElementType>Stream.Create(Adaptor, 'token <it>', <it>);};
  separator="\n"
>
<referencedTokenListLabels
  :{Locals['Stream_<it>'] := TRewriteRule<rewriteElementType>Stream.Create(Adaptor,'token <it>', list_<it>);};
  separator="\n"
>
<referencedRuleLabels:{
if Assigned(<it>) then
  Locals['Stream_<it>'] := TRewriteRuleSubtreeStream.Create(Adaptor, 'token <it>', <it>.Tree)
else
  Locals['Stream_<it>'] := TRewriteRuleSubtreeStream.Create(Adaptor, 'token <it>', nil);}; separator="\n">
<referencedRuleListLabels
  :{Locals['Stream_<it>'] := TRewriteRuleSubtreeStream.Create(Adaptor, 'token <it>', list_<it>);};
  separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
(* <fileName>:<description> *)
if (<referencedElementsDeep:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).HasNext}; separator="
or ">) then
begin
  <alt>
end;
<referencedElementsDeep:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).Reset;<\n>}>
>>

rewriteClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements, // elements in immediately block; no nested blocks
  description) ::=
<<
(* <fileName>:<description> *)
while (<referencedElements:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).HasNext}; separator="
">) do
begin
  <alt>

```

```

end;
<referencedElements:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).Reset();<\n>}>
>>

rewritePositiveClosureBlock(
  alt,rewriteBlockLevel,
  referencedElementsDeep, // all nested refs
  referencedElements,    // elements in immediately block; no nested blocks
  description) ::=
<<
if (not (<referencedElements:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).HasNext}; separator=" or
">)) then
  raise ERewriteEarlyExitException.Create("");

while (<referencedElements:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).HasNext}; separator=" or
">) do
begin
  <alt>
end;
<referencedElements:{el | (Locals['Stream_<el>'] as IRewriteRuleElementStream).Reset();<\n>}>
>>

rewriteAlt(a) ::= <<
(* <a.description> *)
<if(a.pred)>
if (<a.pred>) then
begin
  <a.alt>
end<\n>
<else>
begin
  <a.alt>
end;<\n>
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "Root[0] = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
(* <fileName>:<description> *)
begin
  Root[<treeLevel>] := Adaptor.GetNilNode as I<ASTLabelType>;
  <root:rewriteElement()>
  <children:rewriteElement()>
  Adaptor.AddChild(Root[<enclosingTreeLevel>], Root[<treeLevel>]);
end;<\n>
>>

```



```

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,terminalOptions,args) ::= <<
Adaptor.AddChild(Root[<treeLevel>], <createRewriteNodeFromElement(...)>);<n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
Adaptor.AddChild(Root[<treeLevel>], (Locals['Stream_<label>'] as
IRewriteRuleElementStream).NextNode());<n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
Adaptor.AddChild(Root[<treeLevel>], (Locals['Stream_<label>'] as IRewriteRuleElementStream).NextNode);<n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot((Locals['Stream_<label>'] as
IRewriteRuleElementStream).NextNode(), Root[<treeLevel>]) as I<ASTLabelType>;<n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,terminalOptions,args) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot(<createRewriteNodeFromElement(...)>, Root[<treeLevel>]) as
I<ASTLabelType>;<n>
>>

rewriteImaginaryTokenRef(args,token,terminalOptions,elementIndex) ::= <<
Adaptor.AddChild(Root[<treeLevel>], <createImaginaryNode(tokenType=token, ...)>);<n>
>>

rewriteImaginaryTokenRefRoot(args,token,terminalOptions,elementIndex) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot(<createImaginaryNode(tokenType=token, ...)>, Root[<treeLevel>]) as
I<ASTLabelType>;<n>
>>

```

```

/** plain -> {foo} action */
rewriteAction(action) ::= <<
Root[0] = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
 * let's us refer to $rule to mean previous value. I am reusing the
 * variable 'tree' sitting in retval struct to hold the value of Root[0] right
 * before I set it during rewrites. The assign will be to retval.Tree.
 */
prevRuleRootRef() ::= "RetVal"

rewriteRuleRef(rule) ::= <<
Adaptor.AddChild(Root[<treeLevel>], (Locals['Stream_<rule>'] as IRewriteRuleElementStream).NextTree());<\n>
>>

rewriteRuleRefRoot(rule) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot((Locals['Stream_<rule>'] as IRewriteRuleElementStream).NextNode,
Root[<treeLevel>]) as I<ASTLabelType>;<\n>
>>

rewriteNodeAction(action) ::= <<
Adaptor.AddChild(Root[<treeLevel>], <action>;<\n>
>>

rewriteNodeActionRoot(action) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot(<action>, Root[<treeLevel>]) as I<ASTLabelType>;<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
Adaptor.AddChild(Root[<treeLevel>], (Locals['Stream_<label>'] as IRewriteRuleElementStream).NextTree());<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
Adaptor.AddChild(Root[<treeLevel>], (Locals['Stream_<label>'] as IRewriteRuleElementStream).NextTree());<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot((Locals['Stream_<label>'] as IRewriteRuleElementStream).NextNode,
Root[<treeLevel>]) as I<ASTLabelType>;<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
Root[<treeLevel>] := Adaptor.BecomeRoot((Locals['Stream_<label>'] as IRewriteRuleElementStream).NextNode,

```

```
Root[<treeLevel>]) as I<ASTLabelType>;<\n>
```

```
>>
```

```
createImaginaryNode(tokenType,terminalOptions,args) ::= <<
```

```
<if(terminalOptions.node)>
```

```
<! new MethodNode(IDLabel, args) !>
```

```
T<terminalOptions.node>.Create(<tokenType><if(args)>, <args; separator=", "><endif>)
```

```
<else>
```

```
Adaptor.CreateNode(<tokenType>, <args; separator=", "><if(!args)>'<tokenType>'<endif>) as I<ASTLabelType>
```

```
<endif>
```

```
>>
```

```
createRewriteNodeFromElement(token,terminalOptions,args) ::= <<
```

```
<if(terminalOptions.node)>
```

```
T<terminalOptions.node>.Create((Locals['Stream_<token>'] as IRewriteRuleElementStream).NextToken<if(args)>,<args; separator=", "><endif>)
```

```
<else>
```

```
<else>
```

```
<if(args)> <! must create new node from old !>
```

```
Adaptor.Create(<token>, <args; separator=", ">)
```

```
<else>
```

```
(Locals['Stream_<token>'] as IRewriteRuleElementStream).NextNode
```

```
<endif>
```

```
<endif>
```

```
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/Delphi/AST.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to normal Java output;

* If ASTs are built, then you'll also get ASTDbg.stg loaded.

*/

@outputFile.imports() ::= <<

<@super.imports(>

import org.antlr.runtime.debug.*;

import java.io.IOException;

>>

@genericParser.members() ::= <<

<if(grammar.grammarIsRoot)>

public static final String[] ruleNames = new String[] {

"invalidRule", <grammar.allImportedRules:{rST | "<rST.name>"}; wrap="\n\t", separator=", ">

};<\n>

<endif>

public static final boolean[] decisionCanBacktrack = new boolean[] {

false, // invalid decision

<grammar.decisions:{d | <d.dfa.hasSynPred; null="false">}; wrap="\n ", separator=", ">

};<\n>

<if(grammar.grammarIsRoot)> <! grammar imports other grammar(s) !>

public int ruleLevel = 0;

public int getRuleLevel() { return ruleLevel; }

public void incRuleLevel() { ruleLevel++; }

public void decRuleLevel() { ruleLevel--; }

<if(profile)>

<ctorForProfilingRootGrammar(>

<else>

<ctorForRootGrammar(>

<endif>

<ctorForPredefinedListener(>

<else><! imported grammar !>

public int getRuleLevel() { return <grammar.delegators:{g| <g.delegateName(>>}.getRuleLevel(); }

public void incRuleLevel() { <grammar.delegators:{g| <g.delegateName(>>}.incRuleLevel(); }

public void decRuleLevel() { <grammar.delegators:{g| <g.delegateName(>>}.decRuleLevel(); }

<ctorForDelegateGrammar(>

<endif>

<if(profile)>

public boolean alreadyParsedRule(IntStream input, int ruleIndex) {

int stopIndex = getRuleMemoization(ruleIndex, input.index());

```

        ((Profiler)dbg).examineRuleMemoization(input, ruleIndex, stopIndex,
        <grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
        return super.alreadyParsedRule(input, ruleIndex);
    }

```

@Override

```

public void memoize(IntStream input,
    int ruleIndex,
    int ruleStartIndex)
{
    ((Profiler)dbg).memoize(input, ruleIndex, ruleStartIndex,
    <grammar.composite.rootGrammar.recognizerName>.ruleNames[ruleIndex]);
    super.memoize(input, ruleIndex, ruleStartIndex);
}

```

<endif>

```

protected boolean evalPredicate(boolean result, String predicate) {
    dbg.semanticPredicate(result, predicate);
    return result;
}

```

>>

ctorForRootGrammar() ::= <<

<! bug: can't use <@super.members()> cut-n-paste instead !>

<! Same except we add port number and profile stuff if root grammar !>

```

public <name>(<inputStreamType> input) {
    this(input, DebugEventSocketProxy.DEFAULT_DEBUGGER_PORT, new RecognizerSharedState());
}

```

```

public <name>(<inputStreamType> input, int port, RecognizerSharedState state) {

```

```

    super(input, state);

```

```

    <parserCtorBody()>

```

```

    <createListenerAndHandshake()>

```

```

    <grammar.directDelegates: {g|<g:delegateName()> = new <g.recognizerName>(input, dbg, this.state,
    this<grammar.delegators: {g|, <g:delegateName()>}>);}; separator="\n">

```

```

    <@finally()>

```

```

}

```

>>

ctorForProfilingRootGrammar() ::= <<

<! bug: can't use <@super.members()> cut-n-paste instead !>

```

public <name>(<inputStreamType> input) {
    this(input, new Profiler(null), new RecognizerSharedState());
}

```

```

public <name>(<inputStreamType> input, DebugEventListener dbg, RecognizerSharedState state) {

```

```

    super(input, dbg, state);

```

```

    Profiler p = (Profiler)dbg;

```

```

    p.setParser(this);

```

```

    <parserCtorBody()>

```

```

    <grammar.directDelegates:

```

```

    {g|<g:delegateName()> = new <g.recognizerName>(input, dbg, this.state, this<grammar.delegators:{g|,
<g:delegateName()>>});}; separator="\n">
    <@finally()>
}
<\n>
>>

```

```

/** Basically we don't want to set any dbg listeners are root will have it. */
ctorForDelegateGrammar() ::= <<
public <name>(<inputStreamType> input, DebugEventListener dbg, RecognizerSharedState
state<grammar.delegators:{g|, <g.recognizerName> <g:delegateName()>>}) {
    super(input, dbg, state);
    <parserCtorBody()>
    <grammar.directDelegates:
    {g|<g:delegateName()> = new <g.recognizerName>(input, this, this.state<grammar.delegators:{g|,
<g:delegateName()>>});}; separator="\n">
    }<\n>
>>

```

```

ctorForPredefinedListener() ::= <<
public <name>(<inputStreamType> input, DebugEventListener dbg) {
    <@superClassRef>super(input, dbg, new RecognizerSharedState());<@end>
    <if(profile)>
        Profiler p = (Profiler)dbg;
        p.setParser(this);
    <endif>
    <parserCtorBody()>
    <grammar.directDelegates:{g|<g:delegateName()> = new <g.recognizerName>(input, dbg, this.state,
this<grammar.delegators:{g|, <g:delegateName()>>});}; separator="\n">
    <@finally()>
    }<\n>
>>

```

```

createListenerAndHandshake() ::= <<
<if(TREE_PARSER)>
DebugEventSocketProxy proxy =
    new DebugEventSocketProxy(this, port, input.getTreeAdaptor());<\n>
<else>
DebugEventSocketProxy proxy =
    new DebugEventSocketProxy(this, port, null);<\n>
<endif>
setDebugListener(proxy);
try {
    proxy.handshake();
}
catch (IOException ioe) {
    reportError(ioe);
}

```

>>

```
@genericParser.superClassName() ::= "Debug<@super.superClassName(>"
```

```
@rule.preamble() ::= <<
try { dbg.enterRule(getGrammarFileName(), "<ruleName>");
if ( getRuleLevel()==0 ) {dbg.commence();}
incRuleLevel();
dbg.location(<ruleDescriptor.tree.line>, <ruleDescriptor.tree.charPositionInLine>);<\n>
>>
```

```
@rule.postamble() ::= <<
dbg.location(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.charPositionInLine>);<\n>
}
finally {
dbg.exitRule(getGrammarFileName(), "<ruleName>");
decRuleLevel();
if ( getRuleLevel()==0 ) {dbg.terminate();}
}<\n>
>>
```

```
@synpred.start() ::= "dbg.beginBacktrack(state.backtracking);"
```

```
@synpred.stop() ::= "dbg.endBacktrack(state.backtracking, success);"
```

```
// Common debug event triggers used by region overrides below
```

```
enterSubRule() ::=
"try { dbg.enterSubRule(<decisionNumber>);<\n>"
```

```
exitSubRule() ::=
"} finally {dbg.exitSubRule(<decisionNumber>);}<\n>"
```

```
enterDecision() ::=
"try { dbg.enterDecision(<decisionNumber>, decisionCanBacktrack[<decisionNumber>]);<\n>"
```

```
exitDecision() ::=
"} finally {dbg.exitDecision(<decisionNumber>);}<\n>"
```

```
enterAlt(n) ::= "dbg.enterAlt(<n>);<\n>"
```

```
// Region overrides that tell various constructs to add debugging triggers
```

```
@block.predecision() ::= "<enterSubRule()><enterDecision()>"
```

```
@block.postdecision() ::= "<exitDecision()>"
```

```
@block.postbranch() ::= "<exitSubRule()>"
```

```

@ruleBlock.predecision() ::= "<enterDecision(>"

@ruleBlock.postdecision() ::= "<exitDecision(>"

@ruleBlockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

@blockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

@positiveClosureBlock.preloop() ::= "<enterSubRule(>"

@positiveClosureBlock.postloop() ::= "<exitSubRule(>"

@positiveClosureBlock.predecision() ::= "<enterDecision(>"

@positiveClosureBlock.postdecision() ::= "<exitDecision(>"

@positiveClosureBlock.earlyExitException() ::=
"dbg.recognitionException(eee);<n>"

@closureBlock.preloop() ::= "<enterSubRule(>"

@closureBlock.postloop() ::= "<exitSubRule(>"

@closureBlock.predecision() ::= "<enterDecision(>"

@closureBlock.postdecision() ::= "<exitDecision(>"

@altSwitchCase.prealt() ::= "<enterAlt(altNum)>" // altNum is arg of altSwitchCase

@element.prematch() ::=
"dbg.location(<e.line>,<e.pos>);" // e is arg of element

@matchSet.mismatchedSetException() ::=
"dbg.recognitionException(mse);"

@dfaState.noViableAltException() ::= "dbg.recognitionException(nvae);"

@dfaStateSwitch.noViableAltException() ::= "dbg.recognitionException(nvae);"

dfaDecision(decisionNumber,description) ::= <<
try {
isCyclicDecision = true;
<super.dfaDecision(...)>
}
catch (NoViableAltException nvae) {
dbg.recognitionException(nvae);
throw nvae;
}

```



```
}  
>>
```

```
@cyclicDFA.errorMethod() ::= <<  
public void error(NoViableAltException nvae) {  
    dbg.recognitionException(nvae);  
}  
>>
```

```
/** Force predicate validation to trigger an event */  
evalPredicate(pred,description) ::= <<  
evalPredicate(<pred>,"<description>")  
>>
```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Java/Dbg.stg
No license file was found, but licenses were detected in source scan.

```
/*  
[The "BSD license"]  
Copyright (c) 2008 Erik van Bilzen  
Copyright (c) 2007-2008 Johannes Luber  
Copyright (c) 2005-2007 Kunle Odutola  
Copyright (c) 2005-2006 Terence Parr  
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```

/** Templates for building ASTs during normal parsing.
*
* Deal with many combinations. Dimensions are:
* Auto build or rewrite
* no label, label, list label (label/no-label handled together)
* child, root
* token, set, rule, wildcard
*
* The situation is not too bad as rewrite (->) usage makes ^ and !
* invalid. There is no huge explosion of combinations.
*/
group ASTParser;

@rule.setErrorReturnValue() ::= <<
RetVal.Tree := Adaptor.ErrorNode(Input, RetVal.Start as IToken,
Input.LT(-1), RE) as I<ASTLabelType>;
>>

// TOKEN AST STUFF

/** ID and output=AST */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<super.tokenRef(...)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin<\n>
<endif>
<label>_tree := <createNodeFromToken(...)>;
Adaptor.AddChild(Root[0], <label>_tree);
<if(backtracking)>
end;
<endif>
>>

/** ID! and output=AST (same as plain tokenRef) */
tokenRefBang(token,label,elementIndex) ::= "<super.tokenRef(...)>"

/** ID^ and output=AST */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
<super.tokenRef(...)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin
<endif>
<label>_tree := <createNodeFromToken(...)>;
Root[0] := Adaptor.BecomeRoot(<label>_tree, Root[0]) as I<ASTLabelType>;
<if(backtracking)>

```

```

end;
<endif>
>>

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<listLabel(elem=label,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */
tokenRefRuleRootAndListLabel(token,label,terminalOptions,elementIndex) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<super.matchSet(..., postmatchCode={<if(backtracking)>if (State.Backtracking = 0) then
<endif>Adaptor.AddChild(Root[0], <createNodeFromToken(...)>);}>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,postmatchCode) ::= "<super.matchSet(...)>"

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

```

```

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<if(label)>
<label> := Input.LT(1) as I<labelType>;<\n>
<endif>
<super.matchSet(..., postmatchCode={<if(backtracking)>if (State.Backtracking = 0) then <endif>Root[0] :=
Adaptor.BecomeRoot(<createNodeFromToken(...)>, Root[0]) as I<ASTLabelType>;}>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>Adaptor.AddChild(Root[0], <label>.Tree);
>>

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>Root[0] := Adaptor.BecomeRoot(<label>.Tree, Root[0])
as I<ASTLabelType>;
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label+".Tree",...)>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabel(elem=label+".Tree",...)>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(elem=label+".Tree",...)>
>>

// WILDCARD AST

wildcard(label,elementIndex) ::= <<

```

```

<super.wildcard(...)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin
<endif>
<label>_tree := Adaptor.CreateNode(<label>) as I<ASTLabelType>;
Adaptor.AddChild(Root[0], <label>_tree);
<if(backtracking)>
end;
<endif>
>>

```

```
wildcardBang(label,elementIndex) ::= "<super.wildcard(...)>"
```

```

wildcardRuleRoot(label,elementIndex) ::= <<
<super.wildcard(...)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin
<endif>
<label>_tree := Adaptor.CreateNode(<label>) as I<ASTLabelType>;
Root[0] := Adaptor.BecomeRoot(<label>_tree, Root[0]) as I<ASTLabelType>;
<if(backtracking)>
end;
<endif>
>>

```

```

createNodeFromToken(label,terminalOptions) ::= <<
<if(terminalOptions.node)>
T<terminalOptions.node>.Create(<label>) <! new MethodNode(IDLabel) !>
<else>
Adaptor.CreateNode(<label>) as I<ASTLabelType>
<endif>
>>

```

```

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(backtracking)>
if (State.Backtracking = 0) then
begin<\n>
<endif>
RetVal.Tree := Adaptor.RulePostProcessing(Root[0]) as I<ASTLabelType>;
<if(!TREE_PARSER)>
Adaptor.SetTokenBoundaries(RetVal.Tree, RetVal.Start as IToken, RetVal.Stop as IToken);
<endif>
<if(backtracking)>
<\n>end;
<endif>

```

>>

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Delphi/ASTParser.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC

<http://www.temporal-wave.com>

<http://www.linkedin.com/in/jimidle>

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to normal C output;

* If ASTs are built, then you'll also get ASTDbg.stg loaded.

*/

```
@genericParser.members() ::= <<
```

```
<if(grammar.grammarIsRoot)>
```

```
const char *
```

```
ruleNames[] =
```

```
{
```

```
"invalidRule", <grammar.allImportedRules: {rST | "<rST.name>"}; wrap="\n ", separator=", ">
```

```
};<\n>
```

```
<endif>
```

```

<if(grammar.grammarIsRoot)> <! grammar imports other grammar(s) !>
static ANTLR3_UINT32 ruleLevel = 0;
static ANTLR3_UINT32 getRuleLevel()
{
    return ruleLevel;
}
static void incRuleLevel()
{
    ruleLevel++;
}
static void decRuleLevel()
{
    ruleLevel--;
}
<else> <! imported grammar !>
static ANTLR3_UINT32
getRuleLevel()
{
    return <grammar.delegators:{g| <g:delegateName(>}>->getRuleLevel();
}
static void incRuleLevel()
{
    <grammar.delegators:{g| <g:delegateName(>}>->incRuleLevel();
}
static void
decRuleLevel()
{
    <grammar.delegators:{g| <g:delegateName(>}>.decRuleLevel();
}
<endif>
<if(profile)>
// Profiling not yet implemented for C target
//
<endif>
<if(grammar.grammarIsRoot)>
<ctorForPredefinedListener(>
<else>
<ctorForDelegateGrammar(>
<endif>

static ANTLR3_BOOLEAN
evalPredicate(p<name> ctx, ANTLR3_BOOLEAN result, const char * predicate)
{
    DBG->semanticPredicate(DBG, result, predicate);
    return result;
}<\n>
>>

```

```

@genericParser.debugStuff() ::= <<
<if(grammar.grammarIsRoot)>
<createListenerAndHandshake(>
<endif>
>>

ctorForProfilingRootGrammar() ::= <<
>>

/** Basically we don't want to set any dbg listeners as root will have it. */
ctorForDelegateGrammar() ::= <<

>>

ctorForPredefinedListener() ::= <<

>>

createListenerAndHandshake() ::= <<
{
// DEBUG MODE code
//
pANTLR3_DEBUG_EVENT_LISTENER proxy;
proxy = antlr3DebugListenerNew();
proxy->grammarFileName = INPUT->tokenSource->strFactory->newStr8(INPUT->tokenSource->strFactory,
(pANTLR3_UINT8)ctx->getGrammarFileName());

<if(TREE_PARSER)>
proxy->adaptor = ADAPTOR;
<endif>
PARSER->setDebugListener(PARSER, proxy);

// Try to connect to the debugger (waits forever for a connection)
//
proxy->handshake(proxy);

// End DEBUG MODE code
//
}
>>

@rule.preamble() ::= <<
if ( getRuleLevel()==0 )
{
DBG->commence(DBG);
}
DBG->enterRule(DBG, getGrammarFileName(), (const char *)"<ruleName>");

```



```

incRuleLevel();
DBG->location(DBG, <ruleDescriptor.tree.line>, <ruleDescriptor.tree.column>);<\n>
>>

@rule.postamble() ::= <<
DBG->location(DBG, <ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.column>);<\n>
DBG->exitRule(DBG, getGrammarFileName(), (const char *)"<ruleName>");
decRuleLevel();
if ( getRuleLevel()==0 )
{
  DBG->terminate(DBG);
}
<\n>
>>

@checkRuleBacktrackFailure.debugClean() ::= <<
DBG->exitRule(DBG, getGrammarFileName(), (const char *)"<ruleName>");
decRuleLevel();
>>

@synpred.start() ::= "DBG->beginBacktrack(DBG, BACKTRACKING);"

@synpred.stop() ::= "DBG->endBacktrack(DBG, BACKTRACKING, success);"

// Common debug event triggers used by region overrides below

enterSubRule() ::=
  "DBG->enterSubRule(DBG, <decisionNumber>);<\n>"

exitSubRule() ::=
  "DBG->exitSubRule(DBG, <decisionNumber>);<\n>"

enterDecision() ::=
  "DBG->enterDecision(DBG, <decisionNumber>);<\n>"

exitDecision() ::=
  "DBG->exitDecision(DBG, <decisionNumber>);<\n>"

enterAlt(n) ::= "DBG->enterAlt(DBG, <n>);<\n>"

// Region overrides that tell various constructs to add debugging triggers

@block.predecision() ::= "<enterSubRule()><enterDecision()>"

@block.postdecision() ::= "<exitDecision()>"

@block.postbranch() ::= "<exitSubRule()>"

```

```

@ruleBlock.predecision() ::= "<enterDecision(>"

@ruleBlock.postdecision() ::= "<exitDecision(>"

@ruleBlockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

@blockSingleAlt.prealt() ::= "<enterAlt(n=\"1\")>"

@positiveClosureBlock.preloop() ::= "<enterSubRule(>"

@positiveClosureBlock.postloop() ::= "<exitSubRule(>"

@positiveClosureBlock.predecision() ::= "<enterDecision(>"

@positiveClosureBlock.postdecision() ::= "<exitDecision(>"

@positiveClosureBlock.earlyExitException() ::=
    "DBG->recognitionException(DBG, EXCEPTION);<n>"

@closureBlock.preloop() ::= "<enterSubRule(>"

@closureBlock.postloop() ::= "<exitSubRule(>"

@closureBlock.predecision() ::= "<enterDecision(>"

@closureBlock.postdecision() ::= "<exitDecision(>"

@altSwitchCase.prealt() ::= "<enterAlt(altNum)>"

@element.prematch() ::=
    "DBG->location(DBG, <e.line>, <e.pos>);" // e is arg of element

@matchSet.mismatchedSetException() ::=
    "DBG->recognitionException(DBG, EXCEPTION);"

@newNVEException.noViableAltException() ::= "DBG->recognitionException(DBG, EXCEPTION);"

dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = cdfa<decisionNumber>.predict(ctx, RECOGNIZER, ISTREAM,
&cdfa<decisionNumber>);
if (HASEXCEPTION())
{
    DBG->recognitionException(DBG, EXCEPTION);
    goto rule<ruleDescriptor.name>Ex;
}
<checkRuleBacktrackFailure(>
>>

```

```

@cyclicDFA.errorMethod() ::= <<
//static void
//dfaError(p<name> ctx)
//{{
//  DBG->recognitionException(DBG, EXCEPTION);
//}}
>>

/** Force predicate validation to trigger an event */
evalPredicate(pred,description) ::= <<
evalPredicate(ctx, <pred>, (const char *)"<description>")
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/C/Dbg.stg
```

No license file was found, but licenses were detected in source scan.

```

/*
[The "BSD license"]
Copyright (c) 2007 Kay Roepke 2010 Alan Condit
All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

```

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

/** Templates for building ASTs during tree parsing.
*
* Deal with many combinations. Dimensions are:

```

```

* Auto build or rewrite
* no label, label, list label (label/no-label handled together)
* child, root
* token, set, rule, wildcard
*
* Each combination has its own template except that label/no label
* is combined into tokenRef, ruleRef, ...
*/

/* addition memVars for returnscopes */
@returnScopeInterface.memVars() ::= <<
/* ASTTreeParser returnScopeInterface.memVars */
<recognizer.ASTLabelType; null="CommonTree"> *tree;
>>

/** the interface of returnScope methodsDecl */
@returnScopeInterface.methodsDecl() ::= <<
/* ASTTreeParser returnScopeInterface.methodsDecl */
- (<recognizer.ASTLabelType; null="CommonTree"> *)getTree;
- (void) setTree:(<recognizer.ASTLabelType; null="CommonTree"> *)aTree;<\n>
>>

/** the implementation of returnScope methods */
@returnScope.methods() ::= <<
/* ASTTreeParser returnScope.methods */
- (<ASTLabelType> *)getTree
{
    return tree;
}

- (void) setTree:(<ASTLabelType> *)aTree
{
    if (tree != aTree) {
        if (tree != nil) [tree release];
        if (aTree != nil) [aTree retain];
        tree = aTree;
    }
}

- (void) dealloc
{
    [self setTree:nil];
    [super dealloc];
}

@synthesize tree;
>>

```



```

<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( [input LA:1] == TokenTypeDOWN ) {
  [self match:input TokenType:TokenTypeDOWN Follow:nil]; <checkRuleBacktrackFailure()>
  <children:element()>
  [self match:input TokenType:TokenTypeUP Follow:nil]; <checkRuleBacktrackFailure()>
}
<else>
[self match:input TokenType:TokenTypeDOWN Follow:nil]; <checkRuleBacktrackFailure()>
<children:element()>
[self match:input TokenType:TokenTypeUP Follow:nil]; <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
[treeAdaptor addChild:root_<treeLevel> toTree:root_<enclosingTreeLevel>];
<endif>
_last = _save_last_<treeLevel>;
}<\n>
>>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions) ::= <<
/* ASTTreeParser tokenRefBang */
_last = (<ASTLabelType> *)[input LT:1];
<super.tokenRef(...)>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
/* ASTTreeParser tokenRef */
_last = (<ASTLabelType> *)[input LT:1];
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
  <label>_tree = [<terminalOptions.node> new<terminalOptions.node>:<label>];
<else>
  <label>_tree = (<ASTLabelType> *)[treeAdaptor dupNode:<label>];
<endif><\n>
  [treeAdaptor addChild:<label>_tree toTree:root_<treeLevel>];
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==nil ) _first_<treeLevel> = <label>;
<endif>

```

>>

```
/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
/* ASTTreeParser tokenRefAndListLabel */
<tokenRef(...)>
<listLabel(elem=label,...)>
>>
```

```
/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex) ::= <<
/* ASTTreeParser tokenRefRuleRoot */
_last = (<ASTLabelType> *)[input LT:1];
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = [<terminalOptions.node> new<terminalOptions.node>:<label>];
<else>
<label>_tree = (<ASTLabelType> *)[treeAdaptor dupNode:<label>];
<endif><\n>
root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:<label>_tree old:root_<treeLevel>];
<if(backtracking)>}<endif>
<endif>
>>
```

```
/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
/* ASTTreeParser tokenRefRuleRootAndListLabel */
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>
```

```
/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions) ::= <<
/* ASTTreeParser wildcard */
_last = (<ASTLabelType> *)[input LT:1];
<super.wildcard(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType> *)[adaptor dupTree:<label>];
[adaptor addChild:<label>_tree toTree:root_<treeLevel>];
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel> == nil ) _first_<treeLevel> = <label>;
<endif>
>>
```

```

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
/* ASTTreeParser matchSet */
_last = (<ASTLabelType> *)[input LT:1];
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = [<terminalOptions.node> new<terminalOptions.node>:<label>];
<else>
<label>_tree = (<ASTLabelType> *)[adaptor dupNode:<label>];
<endif><\n>
[adaptor addChild:<label>_tree toTree:root_<treeLevel>];
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
/* ASTTreeParser matchRuleBlockSet */
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
>>

matchSetBang(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
/* ASTTreeParser matchSetBang */
_last = (<ASTLabelType> *)[input LT:1];
<super.matchSet(...)>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
/* ASTTreeParser matchSetRuleRoot */
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = [<terminalOptions.node> new<terminalOptions.node>:<label>];
<else>
<label>_tree = (<ASTLabelType> *)[adaptor dupNode:<label>];
<endif><\n>
root_<treeLevel> = (<ASTLabelType> *)[adaptor becomeRoot:<label>_tree old:root_<treeLevel>];
<if(backtracking)>\}<endif>
<endif>
}, ...
)>

```


>>

// RULE REF AST

/** rule auto construct */

ruleRef(rule,label,elementIndex,args,scope) ::= <<

/* ASTTreeParser ruleRef */

_last = (<ASTLabelType> *)[input LT:1];

<super.ruleRef(...)>

<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>

<if(!rewriteMode)>

[treeAdaptor addChild:<label>.tree toTree:root_<treeLevel>];

<else> <! rewrite mode !>

if (_first_<treeLevel> == nil) _first_<treeLevel> = <label>.tree;

<endif>

>>

/** x+=rule auto construct */

ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<

/* ASTTreeParser ruleRefAndListLabel */

<ruleRef(...)>

<! <listLabel(elem = "["+label+" getTree]",...)> !>

<listLabel(elem = {[<label> getTree]},...)>

>>

/** ^(rule ...) auto construct */

ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<

/* ASTTreeParser ruleRefRuleRoot */

_last = (<ASTLabelType> *)[input LT:1];

<super.ruleRef(...)>

<if(!rewriteMode)>

<if(backtracking)>if (state.backtracking == 0) <endif>

root_<treeLevel> = (<ASTLabelType> *)[treeAdaptor becomeRoot:<label>.tree old:root_<treeLevel>];

<endif>

>>

/** ^(x+=rule ...) auto construct */

ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<

/* ASTTreeParser ruleRefRuleRootAndListLabel */

<ruleRefRuleRoot(...)>

<listLabel(elem = {[<label> getTree]},...)>

>>

/** rule when output=AST and tracking for rewrite */

ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<

/* ASTTreeParser ruleRefTrack */

_last = (<ASTLabelType> *)[input LT:1];

<super.ruleRefTrack(...)>

>>

```
/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ASTTreeParser ruleRefTrackAndListLabel */
_last = (<ASTLabelType> *)[input LT:1];
<super.ruleRefTrackAndListLabel(...)>
>>
```

```
/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
/* ASTTreeParser ruleRefRuleRootTrack */
_last = (<ASTLabelType> *)[input LT:1];
<super.ruleRefRuleRootTrack(...)>
>>
```

```
/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
/* ASTTreeParser ruleRefRuleRootTrackAndListLabel */
_last = (<ASTLabelType> *)[input LT:1];
<super.ruleRefRuleRootTrackAndListLabel(...)>
>>
```

```
/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,terminalOptions,scope) ::= <<
/* ASTTreeParser createRewriteNodeFromElement */
<if(terminalOptions.node)>
<! new <terminalOptions.node>(stream_<token>.nextNode()) !>
[[[<terminalOptions.node>(stream_<token> alloc) init] nextNode];
<else>
<! stream_<token>.nextNode() !>
[stream_<token> nextNode]
<endif>
>>
```

```
ruleCleanUp() ::= <<
/* ASTTreeParser ruleCleanUp */
<super.ruleCleanUp()>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<\n><endif>
retval.tree = (<ASTLabelType> *)[treeAdaptor rulePostProcessing:root_0];
<if(backtracking)>}<endif>
<endif>
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/ObjC/ASTTreeParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2007-2008 Johannes Luber
```

```
* Copyright (c) 2005-2007 Kunle Odutola
```

```
* Copyright (c) 2005 Terence Parr
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

```
* modification, are permitted provided that the following conditions
```

```
* are met:
```

```
* 1. Redistributions of source code must retain the above copyright
```

```
* notice, this list of conditions and the following disclaimer.
```

```
* 2. Redistributions in binary form must reproduce the above copyright
```

```
* notice, this list of conditions and the following disclaimer in the
```

```
* documentation and/or other materials provided with the distribution.
```

```
* 3. The name of the author may not be used to endorse or promote products
```

```
* derived from this software without specific prior written permission.
```

```
*
```

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```

```
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

```
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

```
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
```

```
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
```

```
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

```
/** Template overrides to add debugging to AST stuff. Dynamic inheritance
```

```
* hierarchy is set up as ASTDbg : AST : Dbg : Java by code generator.
```

```
*/
```

```
parserMembers() ::= <<
```

```
protected DebugTreeAdaptor adaptor;
```

```
public ITreeAdaptor TreeAdaptor
```

```
{
```

```
get
```

```
{
```

```
return adaptor;
```

```
}
```

```
set
```

```

    {
    <if(grammar.grammarIsRoot)>
        this.adaptor = new DebugTreeAdaptor(dbg,adaptor);
    <else>
        this.adaptor = (DebugTreeAdaptor)adaptor; // delegator sends dbg adaptor
    <endif><\n>
        <grammar.directDelegates:{g|<g:delegateName()>.TreeAdaptor = this.adaptor;}>
    }
    }<\n>
>>

```

```

parserCtorBody() ::= <<
<super.parserCtorBody()>
>>

```

```

createListenerAndHandshake() ::= <<
DebugEventSocketProxy proxy = new DebugEventSocketProxy( this, port,
<if(TREE_PARSER)>input.TreeAdaptor<else>adaptor<endif> );
DebugListener = proxy;
<inputStreamType> = new Debug<inputStreamType>( input, proxy );
try
{
    proxy.Handshake();
}
catch ( IOException ioe )
{
    ReportError( ioe );
}
>>

```

```

@ctorForRootGrammar.finally() ::= <<
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;
proxy.TreeAdaptor = adap;
>>

```

```

@ctorForProfilingRootGrammar.finally() ::= <<
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;
>>

```

```

@ctorForPredefinedListener.superClassRef() ::= ": base( input, dbg )"

```

```

@ctorForPredefinedListener.finally() ::= <<
<if(grammar.grammarIsRoot)><! don't create new adaptor for delegates !>
ITreeAdaptor adap = new CommonTreeAdaptor();
TreeAdaptor = adap;<\n>
<endif>

```

>>

```
//@rewriteElement.pregen() ::= "dbg.Location( <e.line>, <e.pos> );"
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/CSharp2/ASTDbg.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
* [The "BSD license"]
```

```
* Copyright (c) 2005-2008 Terence Parr
```

```
* All rights reserved.
```

```
*
```

```
* Conversion to C#:
```

```
* Copyright (c) 2008-2010 Sam Harwell, Pixel Mine, Inc.
```

```
* All rights reserved.
```

```
*
```

```
* Redistribution and use in source and binary forms, with or without
```

```
* modification, are permitted provided that the following conditions
```

```
* are met:
```

```
* 1. Redistributions of source code must retain the above copyright
```

```
* notice, this list of conditions and the following disclaimer.
```

```
* 2. Redistributions in binary form must reproduce the above copyright
```

```
* notice, this list of conditions and the following disclaimer in the
```

```
* documentation and/or other materials provided with the distribution.
```

```
* 3. The name of the author may not be used to endorse or promote products
```

```
* derived from this software without specific prior written permission.
```

```
*
```

```
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```

```
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
```

```
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
```

```
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
```

```
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
```

```
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
```

```
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
```

```
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
```

```
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
```

```
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
```

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/CSharp3Target.java
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2009 Gokulakannan Somasundaram,
```

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

* This code generating template and the associated Cpp runtime was produced by:

* Gokulakannan Somasundaram (heavy lifting from C Run-time by Jim Idle)

*/

```
cTypeInitMap ::= [  
  "int"   : "0",      // Integers  start out being 0  
  "long"  : "0",      // Longs    start out being 0  
  "float" : "0.0",    // Floats  start out being 0  
  "double": "0.0",    // Doubles start out being 0  
  "bool"  : "false", // Booleans start out being Antlr C for false  
  "byte"  : "0",      // Bytes   start out being 0  
  "short" : "0",      // Shorts  start out being 0  
  "char"  : "0"       // Chars   start out being 0  
]
```

```
leadIn(type) ::=
```

```
<<
```

```
/** \file
```

```
* This <type> file was generated by $ANTLR version <ANTLRVersion>
```

```
*
```

```
* - From the grammar source file : <fileName>
```

```
* - On : <generatedTimestamp>
```

```

<if(LEXER)>
*   -           for the lexer : <name>Lexer
<endif>
<if(PARSER)>
*   -           for the parser : <name>Parser
<endif>
<if(TREE_PARSER)>
*   -           for the tree parser : <name>TreeParser
<endif>
*
* Editing it, at least manually, is not wise.
*
* C++ language generator and runtime by Gokulakannan Somasundaram ( heavy lifting from C Run-time by Jim
Idle )
*
*
>>

/** The overall file structure of a recognizer; stores methods for rules
* and cyclic DFAs plus support code.
*/
outputFile( LEXER,
            PARSER,
            TREE_PARSER,
            actionScope,
            actions,
            docComment,
            recognizer,
            name,
            tokens,
            tokenNames,
            rules,
            cyclicDFAs,
            bitsets,
            buildTemplate,
            buildAST,
            rewriteMode,
            profile,
            backtracking,
            synpreds,
            memoize,
            numRules,
            fileName,
            ANTLRVersion,
            generatedTimestamp,
            trace,
            scopes,
            superClass,

```

```

        literals
    ) ::=
<<
<leadIn("C++ source")>
*/
// [The "BSD license"]
// Copyright (c) 2005-2009 Gokulakannan Somasundaram, ElectronDB
//
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. The name of the author may not be used to endorse or promote products
// derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
// IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
// OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
// IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
// NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
// THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

<if(actions.(actionScope).header)>

/* =====
* This is what the grammar programmer asked us to put at the top of every file.
*/
<actions.(actionScope).header>
/* End of Header action.
* =====
*/
<endif>

/* -----
* Include the ANTLR3 generated header file.
*/
#include "<name>.hpp"
<if(trace)>

```



```

#include <iostream>
<endif>
<if(recognizer.grammar.delegators)>
// Include delegator definition header files
//
<recognizer.grammar.delegators: {g#include "<g.recognizerName>.hpp" }; separator="\n">
<endif>

<actions.(actionScope).postinclude>
/* ----- */

<docComment>

<if(literals)>

<beginNamespace(actions)>

/** String literals used by <name> that we must do things like MATCHS() with.
 * C will normally just lay down 8 bit characters, and you can use L"xxx" to
 * get wchar_t, but wchar_t is 16 bits on Windows, which is not UTF32 and so
 * we perform this little trick of defining the literals as arrays of UINT32
 * and passing in the address of these.
 */
<literals:{it | static ANTLR_UCHAR lit_<i>[] = <it>;}; separator="\n">

<endNamespace(actions)>

<endif>

/* ===== */

/* =====
 * Start of recognizer
 */

<recognizer>

/* End of code
 * =====
 */

>>
headerFileExtension() ::= ".hpp"

beginNamespace(actions) ::= <%
<if(actions.(actionScope).namespace)>
namespace <actions.(actionScope).namespace> {
<endif>

```

```
%>
```

```
endNamespace(actions) ::= <%  
<if(actions.(actionScope).namespace)>  
}  
<endif>  
%>
```

```
headerFile( LEXER,  
    PARSER,  
    TREE_PARSER,  
    actionScope,  
    actions,  
    docComment,  
    recognizer,  
    name,  
    tokens,  
    tokenNames,  
    rules,  
    cyclicDFAs,  
    bitsets,  
    buildTemplate,  
    buildAST,  
    rewriteMode,  
    profile,  
    backtracking,  
    synpreds,  
    memoize,  
    numRules,  
    fileName,  
    ANTLRVersion,  
    generatedTimestamp,  
    trace,  
    scopes,  
    superClass,  
    literals  
    ) ::=  
<<  
<leadIn("C++ header")>  
<if(PARSER)>  
* The parser <mainName()> has the callable functions (rules) shown below,  
<endif>  
<if(LEXER)>  
* The lexer <mainName()> has the callable functions (rules) shown below,  
<endif>  
<if(TREE_PARSER)>  
* The tree parser <mainName()> has the callable functions (rules) shown below,
```

```

<endif>
* which will invoke the code for the associated rule in the source grammar
* assuming that the input stream is pointing to a token/text stream that could begin
* this rule.
*
* For instance if you call the first (topmost) rule in a parser grammar, you will
* get the results of a full parse, but calling a rule half way through the grammar will
* allow you to pass part of a full token stream to the parser, such as for syntax checking
* in editors and so on.
*
*/
// [The "BSD license"]
// Copyright (c) 2005-2009 Gokulakannan Somasundaram.
//
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. The name of the author may not be used to endorse or promote products
// derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
// IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
// OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
// IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
// NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
// THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

#ifndef _<name>_H
#define _<name>_H
<actions.(actionScope).preincludes>
/* =====
* Standard antlr3 C++ runtime definitions
*/
#include \<antlr3.hpp>

/* End of standard antlr 3 runtime definitions
* =====

```

```

*/

<actions.(actionScope).includes>

<if(recognizer.grammar.delegates)>
// Include delegate definition header files
//
<recognizer.grammar.delegates: {g|#include "<g.recognizerName>.hpp"}; separator="\n">

<endif>

<actions.(actionScope).header>

#ifdef WIN32
// Disable: Unreferenced parameter, - Rules with parameters that are not used
// constant conditional, - ANTLR realizes that a prediction is always true (synpred usually)
// initialized but unused variable - tree rewrite variables declared but not needed
// Unreferenced local variable - lexer rule declares but does not always use _type
// potentially uninitialized variable used - retval always returned from a rule
// unreferenced local function has been removed - susually getTokenNames or freeScope, they can go without
warnigns
//
// These are only really displayed at warning level /W4 but that is the code ideal I am aiming at
// and the codegen must generate some of these warnings by necessity, apart from 4100, which is
// usually generated when a parser rule is given a parameter that it does not use. Mostly though
// this is a matter of orthogonality hence I disable that one.
//
#pragma warning( disable : 4100 )
#pragma warning( disable : 4101 )
#pragma warning( disable : 4127 )
#pragma warning( disable : 4189 )
#pragma warning( disable : 4505 )
#pragma warning( disable : 4701 )
#endif
<if(backtracking)>

/* =====
* BACKTRACKING IS ENABLED
* =====
*/
<endif>

<beginNamespace(actions)>

<if(recognizer.grammar.delegators)>
// Include delegator definition classes
//

```

```

<recognizer.grammar.delegates: {g|class <g.recognizerName>; }; separator="\n">
<endif>

<actions.(actionScope).traits>
typedef <name>Traits <name>ImplTraits;

<rules:{r | <if(r.ruleDescriptor.isSynPred)> struct <r.ruleDescriptor.name> {\}; <endif>}; separator="\n">

class <name>Tokens
{
public:
/** Symbolic definitions of all the tokens that the <grammarType()> will work with.
*
* Antlr will define EOF, but we can't use that as it is too common in
* in C header files and that would be confusing. There is no way to filter this out at the moment
* so we just undef it here for now. That isn't the value we get back from C recognizers
* anyway. We are looking for ANTLR_TOKEN_EOF.
*/
enum Tokens
{
EOF_TOKEN = <name>ImplTraits::CommonTokenType::TOKEN_EOF
<tokens:{it | , <it.name> = <it.type> }; separator="\n">
};

/** Context tracking structure for <mainName()>
*/
class <name> : public <componentBaseType()>, public <name>Tokens
{
public:
typedef <name>ImplTraits ImplTraits;
typedef <name> ComponentType;
typedef ComponentType::StreamType StreamType;
typedef <componentBaseType()> BaseType;
typedef ImplTraits::RecognizerSharedStateType\<StreamType> RecognizerSharedStateType;
typedef StreamType InputType;
<if(recognizer.filterMode)>
static const bool IsFiltered = true;
<else>
static const bool IsFiltered = false;
<endif>

<scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeDecl(it)><endif> }>
<rules:{r | <if(r.ruleDescriptor.ruleScope)><ruleAttributeScopeDecl(scope=r.ruleDescriptor.ruleScope)><endif> }>

private:
<if(recognizer.grammar.delegates)>

```

```

<recognizer.grammar.delegates:
  {g|<g.recognizerName>* m_<g.delegateName()>;}; separator="\n">
<endif>
<if(recognizer.grammar.delegators)>
  <recognizer.grammar.delegators:
    {g|<g.recognizerName>* m_<g.delegateName()>;}; separator="\n">
  <endif>
  <scopes:{it | <if(it.isDynamicGlobalScope)>
    <globalAttributeScopeDef(it)>
  <endif>}; separator="\n\n">
  <rules: {r |<if(r.ruleDescriptor.ruleScope)>
    <ruleAttributeScopeDef(scope=r.ruleDescriptor.ruleScope)>
  <endif>}>
  <@members>
  <@end>

public:
  <name>(InputType* instream<recognizer.grammar.delegators:{g|, <g.recognizerName>*
<g.delegateName()>}>);
  <name>(InputType* instream, RecognizerSharedStateType* state<recognizer.grammar.delegators:{g|,
<g.recognizerName>* <g.delegateName()>}>);

  void init(InputType* instream <recognizer.grammar.delegators:{g|, <g.recognizerName>* <g.delegateName()>}>
);

  <actions.(actionScope).context>

<if(LEXER)>
<if(recognizer.filterMode)>
  void memoize(ANTLR_MARKER ruleIndex, ANTLR_MARKER ruleParseStart);
  bool alreadyParsedRule(ANTLR_MARKER ruleIndex);
  <filteringNextToken()>
<endif>
  <rules:{r | <if(!r.ruleDescriptor.isSynPred)><headerReturnType(ruleDescriptor=r.ruleDescriptor)>
m<r.ruleDescriptor.name>( <r.ruleDescriptor.parameterScope:parameterScope()>);<endif>}; separator="\n">
  <rules:{r | <if(r.ruleDescriptor.isSynPred)> <headerReturnType(ruleDescriptor=r.ruleDescriptor)> msynpred(
antlr3::ClassForwarder< <r.ruleDescriptor.name> > <r.ruleDescriptor.parameterScope:parameterScope()>);
  void m<r.ruleDescriptor.name>_fragment (<r.ruleDescriptor.parameterScope:parameterScope()>);<endif>};
separator="\n">
<endif>
<if(!LEXER)>
  <rules:{r | <headerReturnScope(ruleDescriptor=r.ruleDescriptor)>}>
  <rules:{r | <if(!r.ruleDescriptor.isSynPred)> <headerReturnType(ruleDescriptor=r.ruleDescriptor)>
<r.ruleDescriptor.name> (<r.ruleDescriptor.parameterScope:parameterScope()>); <endif>}; separator="\n">
  <rules:{r | <if(r.ruleDescriptor.isSynPred)> <headerReturnType(ruleDescriptor=r.ruleDescriptor)> msynpred(
antlr3::ClassForwarder< <r.ruleDescriptor.name> > <r.ruleDescriptor.parameterScope:parameterScope()>);
  void m<r.ruleDescriptor.name>_fragment (<r.ruleDescriptor.parameterScope:parameterScope()>);<endif>};
separator="\n">

```

```

<! generate rule/method definitions for imported rules so they
appear to be defined in this recognizer. !>
// Delegated rules
<recognizer.grammar.delegatedRules:{ruleDescriptor|
  <headerReturnType(ruleDescriptor)>
  <ruleDescriptor.name>(<ruleDescriptor.parameterScope:parameterScope()>);}; separator="\n">
<endif>

const char *  getGrammarFileName();
void          reset();
~<name>();

};

// Function prototypes for the constructor functions that external translation units
// such as delegators and delegates may wish to call.
//
<if(!recognizer.grammar.grammarIsRoot)>
extern ANTLR_UINT8*  <recognizer.grammar.composite.rootGrammar.recognizerName>TokenNames[];
<endif>

/* End of token definitions for <name>
* =====
*/

<endNamespace(actions)>

#endif

/* END - Note:Keep extra line feed to satisfy UNIX systems */

>>

grammarType() ::= <%
<if(PARSER)>
parser
<endif>
<if(LEXER)>
lexer
<endif>
<if(TREE_PARSER)>
tree parser
<endif>
%>

componentType() ::= <<
<if(PARSER)>

```

```
<name>ImplTraits::ParserType
<endif>
<if(LEXER)>
<name>ImplTraits::LexerType
<endif>
<if(TREE_PARSER)>
<name>ImplTraits::TreeParserType
<endif>
>>
```

```
componentBaseType() ::= <%
<if(PARSER)>
<name>ImplTraits::BaseParserType
<endif>
<if(LEXER)>
<name>ImplTraits::BaseLexerType
<endif>
<if(TREE_PARSER)>
<name>ImplTraits::BaseTreeParserType
<endif>
%>
```

```
streamType() ::= <<
<if(PARSER)>
<name>ImplTraits::ParserType::StreamType
<endif>
<if(LEXER)>
<name>ImplTraits::LexerType::StreamType
<endif>
<if(TREE_PARSER)>
<name>ImplTraits::TreeParserType::StreamType
<endif>
>>
```

```
mainName() ::= <%
<if(PARSER)>
<name>
<endif>
<if(LEXER)>
<name>
<endif>
<if(TREE_PARSER)>
<name>
<endif>
%>
```

```
headerReturnScope(ruleDescriptor) ::= "<returnScope(scope=ruleDescriptor.returnScope)>"
```



```

headerReturnType(ruleDescriptor) ::= <%
<if(LEXER)>
<if(!ruleDescriptor.isSynPred)>
void
<else>
<returnType()>
<endif>
<else>
<returnType()>
<endif>
%>

// Produce the lexer output
//
lexer( grammar,
name,
tokens,
scopes,
rules,
numRules,
filterMode,
superClass,
labelType="ImplTraits::CommonTokenType*" ) ::= <<

using namespace antlr3;

<beginNamespace(actions)>

<if(filterMode)>

/* Override the normal MEMOIZE and HAVEALREADYPARSED macros as this is a filtering
* lexer. In filter mode, the memoizing and backtracking are gated at BACKTRACKING > 1 rather
* than just BACKTRACKING. In some cases this might generate code akin to:
* if (BACKTRACKING) if (BACKTRACKING > 1) memoize.
*/
void <name>::memoize(ANTLR_MARKER ruleIndex, ANTLR_MARKER ruleParseStart)
{
BaseType* base = this;
if ( this->get_backtracking()>1 )
base->memoize( ruleIndex, ruleParseStart );
}

bool <name>::alreadyParsedRule(ANTLR_MARKER ruleIndex)
{
BaseType* base = this;
if ( this->get_backtracking() > 1 )

```

```

    return base->haveParsedRule(ruleIndex);
return false;
}

<endif>

/* =====
* Lexer matching rules end.
* =====
*/

<scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>

<actions.lexer.members>

<name>::~~<name>()
{
<if(memoize)>
RuleMemoType* rulememo = this->getRuleMemo();
if(rulememo != NULL)
{
delete rulememo;
this->setRuleMemo(NULL);
}
<endif>
<if(grammar.directDelegates)>
// Free the lexers that we delegated to
// functions to. NULL the state so we only free it once.
//
<grammar.directDelegates:
    {g| m_<g:delegateName()->set_lexstate(NULL);
    delete m_<g:delegateName(>; }; separator="\n">
<endif>
}

void
<name>::~reset()
{
    this->get_rec()->reset();
}

/** \brief Name of the grammar file that generated this code
*/
static const char fileName[] = "<fileName>";

/** \brief Return the name of the grammar file that generated this code.
*/
const char* <name>::getGrammarFileName()

```

```

{
return fileName;
}

/** \brief Create a new lexer called <name>
 *
 * \param[in] instream Pointer to an initialized input stream
 * \return
 * - Success p<name> initialized for the lex start
 * - Fail NULL
 */
<name>::<name>(StreamType* instream<grammar.delegators:{g|, <g.recognizerName>* <g.delegateName()>>>)
:<name>ImplTraits::BaseLexerType(ANTLR_SIZE_HINT, instream, NULL)
{
// See if we can create a new lexer with the standard constructor
//
this->init(instream <grammar.delegators:{g|, <g.delegateName()>>>);
}

/** \brief Create a new lexer called <name>
 *
 * \param[in] instream Pointer to an initialized input stream
 * \param[state] state Previously created shared recognizer stat
 * \return
 * - Success p<name> initialized for the lex start
 * - Fail NULL
 */
<name>::<name>(StreamType* instream, RecognizerSharedStateType* state<grammar.delegators:{g|,
<g.recognizerName>* <g.delegateName()>>>)
:<name>ImplTraits::BaseLexerType(ANTLR_SIZE_HINT, instream, state)
{
this->init(instream <grammar.delegators:{g|, <g.delegateName()>> >);
}

void <name>::init(StreamType* instream<grammar.delegators:{g|, <g.recognizerName>* <g.delegateName()>> >)
{
/* -----
 * Memory for basic structure is allocated, now to fill in
 * in base ANTLR3 structures. We initialize the function pointers
 * for the standard ANTLR3 lexer function set, but upon return
 * from here, the programmer may set the pointers to provide custom
 * implementations of each function.
 *
 * We don't use the macros defined in <name>.h here so you can get a sense
 * of what goes where.
 */

<if(memoize)>

```

```

<if(grammar.grammarIsRoot)>
  // Create a LIST for recording rule memos.
  //
  this->setRuleMemo( new IntTrie(15) ); /* 16 bit depth is enough for 32768 rules! */
<endif>
<endif>

<if(grammar.directDelegates)>
  // Initialize the lexers that we are going to delegate some
  // functions to.
  //
  <grammar.directDelegates:
    { g|m_<g:delegateName()> = new <g.recognizerName>(instream, this->get_lexstate(),
this<grammar.delegators:{ g|, <g:delegateName()>}>);}; separator="\n">
  <endif>
<if(grammar.delegators)>
  // Install the pointers back to lexers that will delegate us to perform certain functions
  // for them.
  //
  <grammar.delegators:
    { g| m_<g:delegateName()> = <g:delegateName()>;}; separator="\n">
  <endif>
}

<if(cyclicDFAs)>

/* =====
* DFA tables for the lexer
*/
<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>
/* =====
* End of DFA tables for the lexer
*/
<endif>

/* =====
* Functions to match the lexer grammar defined tokens from the input stream
*/

<rules; separator="\n\n">

/* =====
* Lexer matching rules end.
* =====
*/
<if(synpreds)>

/* =====

```

```

* Lexer syntactic predicates
*/
<synpreds: {p | <lexerSynpred(predname=p)>}>
/* =====
* Lexer syntactic predicates end.
* =====
*/
<endif>

/* End of Lexer code
* =====
* =====
*/

<endNamespace(actions)>

>>

filteringNextToken() ::= <<
<name>ImplTraits::CommonTokenType*
<name>ImplTraits::TokenSourceType::nextToken()
{
  LexerType* lexer;
  typename LexerType::RecognizerSharedStateType* state;

  lexer = this->get_super();
  state = lexer->get_lexstate();

  /* Get rid of any previous token (token factory takes care of
   * any deallocation when this token is finally used up.
   */
  state->set_token_present( false );
  state->set_error( false ); /* Start out without an exception */
  state->set_failedflag(false);

  /* Record the start of the token in our input stream.
   */
  state->set_tokenStartCharIndex( lexer->index());
  state->set_tokenStartCharPositionInLine( lexer->getCharPositionInLine() );
  state->set_tokenStartLine( lexer->getLine() );
  state->set_text("");

  /* Now call the matching rules and see if we can generate a new token
   */
  for (;;)
  {
    if (lexer->LA(1) == ANTLR_CHARSTREAM_EOF)

```



```

/** How to generate a parser */
genericParser( grammar, name, scopes, tokens, tokenNames, rules, numRules,
              bitsets, inputStreamType, superClass,
              labelType, members, rewriteElementType,
              filterMode, ASTLabelType="ImplTraits::TreeType*") ::= <<

using namespace antlr3;
<if(grammar.grammarIsRoot)>
/** \brief Table of all token names in symbolic order, mainly used for
 *   error reporting.
 */
ANTLR_UINT8* <name>TokenNames[<length(tokenNames)>+4]
= {
    (ANTLR_UINT8*) "\<invalid>", /* String to print to indicate an invalid token */
    (ANTLR_UINT8*) "\<EOR>",
    (ANTLR_UINT8*) "\<DOWN>",
    (ANTLR_UINT8*) "\<UP>",
    <tokenNames:{it |(ANTLR_UINT8*) <it>}; separator=",\n">
};
<endif>

<@members>

<@end>

/** \brief Name of the grammar file that generated this code
 */
static const char fileName[] = "<fileName>";

/** \brief Return the name of the grammar file that generated this code.
 */
const char* <name>::getGrammarFileName()
{
    return fileName;
}
/** \brief Create a new <name> parser and return a context for it.
 *
 * \param[in] instream Pointer to an input stream interface.
 *
 * \return Pointer to new parser context upon success.
 */
<name>::<name>( StreamType* instream<grammar.delegators:{g|, <g.recognizerName>* <g.delegateName()>}>
<constructorInitializerType("NULL")>
{
    // See if we can create a new parser with the standard constructor
    //
    this->init(instream<grammar.delegators:{g|, <g.delegateName()>}>);
}

```

```

}

/** \brief Create a new <name> parser and return a context for it.
 *
 * \param[in] instream Pointer to an input stream interface.
 *
 * \return Pointer to new parser context upon success.
 */
<name>::<name>( StreamType* instream, RecognizerSharedStateType* state<grammar.delegators:{g|,
<g.recognizerName>* <g.delegateName()>>>)
<constructorInitializerType("state")>
{
  this->init(instream <grammar.delegators:{g|, <g.delegateName()>>});
}

void <name>::init(StreamType* instream<grammar.delegators:{g|, <g.recognizerName>* <g.delegateName()>>})
{
  <actions.parser.apifuncs>
  <if(memoize)>
  <if(grammar.grammarIsRoot)>
    /* Create a LIST for recording rule memos.
    */
    typedef RecognizerSharedStateType::RuleMemoType RuleMemoType;
    this->setRuleMemo( new RuleMemoType(15) ); /* 16 bit depth is enough for 32768 rules! */<\n>
  <endif>
  <endif>
  <if(grammar.directDelegates)>
    // Initialize the lexers that we are going to delegate some
    // functions to.
    //
    <grammar.directDelegates:
      {g|m_<g.delegateName()> = new <g.recognizerName>(instream, this->get_psrstate(),
this-<grammar.delegators:{g|, <g.delegateName()>>}); separator="\n">
    <endif>
    <if(grammar.delegators)>
      // Install the pointers back to lexers that will delegate us to perform certain functions
      // for them.
      //
      <grammar.delegators: {g| m_<g.delegateName()> = <g.delegateName()>; separator="\n">
    <endif>
    /* Install the token table
    */
    this->get_psrstate()->set_tokenNames( <grammar.composite.rootGrammar.recognizerName>TokenNames );

  <@debugStuff()>
}

```



```

void
<name>::reset()
{
    this->get_rec()->reset();
}

/** Free the parser resources
*/
<name>::~~<name>()
{
    <@cleanup>
    <@end>
<if(grammar.directDelegates)>
    // Free the parsers that we delegated to
    // functions to.NULL the state so we only free it once.
    //
    <grammar.directDelegates:
        {g| m_<g:delegateName()->set_psrstate( NULL );
        delete m_<g:delegateName()->; separator="\n">
    <endif>
<if(memoize)>
<if(grammar.grammarIsRoot)>
    if(this->getRuleMemo() != NULL)
    {
        delete this->getRuleMemo();
        this->setRuleMemo(NULL);
    }
    <endif>
<endif>
}

/** Return token names used by this <grammarType>
*
* The returned pointer is used as an index into the token names table (using the token
* number as the index).
*
* \return Pointer to first char * in the table.
*/
static ANTLR_UINT8** getTokenNames()
{
    return <grammar.composite.rootGrammar.recognizerName>TokenNames;
}

<members>

/* Declare the bitsets
*/
<bitsets:{it | <bitsetDeclare(bitsetName={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},

```

```
words64=it.bits, traits={ <name>ImplTraits } )>>
```

```
<if(cyclicDFAs)>
```

```
/* =====
```

```
* DFA tables for the parser
```

```
*/
```

```
<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>
```

```
/* =====
```

```
* End of DFA tables for the parser
```

```
*/
```

```
<endif>
```

```
/* =====
```

```
* Parsing rules
```

```
*/
```

```
<rules; separator="\n\n">
```

```
<if(grammar.delegatedRules)>
```

```
// Delegated methods that appear to be a part of this
```

```
// parser
```

```
//
```

```
<grammar.delegatedRules:{ruleDescriptor|
```

```
  <return Type()> <name>::<ruleDescriptor.name>( <ruleDescriptor.parameterScope:parameterScope()>)
```

```
  {
```

```
    <if(ruleDescriptor.hasReturnValue)>return <endif>m_<ruleDescriptor.grammar:delegateName()>-
```

```
><ruleDescriptor.name>( <if(ruleDescriptor.parameterScope)><ruleDescriptor.parameterScope.attributes:{ a|<a.name>  
>}; separator="," ><endif>);
```

```
  \}}; separator="\n">
```

```
<endif>
```

```
/* End of parsing rules
```

```
* =====
```

```
*/
```

```
/* =====
```

```
* Syntactic predicates
```

```
*/
```

```
<synpreds:{ p | <synpred(predname=p)> }>
```

```
/* End of syntactic predicates
```

```
* =====
```

```
*/
```

```
>>
```

```
constructorInitializerType(rec_state) ::= <<
```

```
<if(PARSER)>
```

```
  :ImplTraits::BaseParserType(ANTLR_SIZE_HINT, instream, <rec_state>)
```

```

<endif>
<if(TREE_PARSER)>
  :ImplTraits::BaseTreeParserType(ANTLR_SIZE_HINT, instream, <rec_state>)
<endif>
>>

parser( grammar,
  name,
  scopes,
  tokens,
  tokenNames,
  rules,
  numRules,
  bitsets,
  ASTLabelType,
  superClass="Parser",
  labelType="ImplTraits::CommonTokenType*",
  members={<actions.parser.members>}
) ::= <<
<beginNamespace(actions)>
<genericParser(inputStreamType="CommonTokenStreamType*", rewriteElementType="Token", filterMode=false,
...)>
<endNamespace(actions)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser( grammar,
  name,
  scopes,
  tokens,
  tokenNames,
  globalAction,
  rules,
  numRules,
  bitsets,
  filterMode,
  labelType={<ASTLabelType>},
  ASTLabelType="ImplTraits::TreeType*",
  superClass="TreeParser",
  members={<actions.treeparser.members>}
) ::= <<
<beginNamespace(actions)>
<genericParser(inputStreamType="CommonTreeNodeStream*", rewriteElementType="Node", ...)>
<endNamespace(actions)>
>>

```

```

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start <ruleName>
void <name>::m<ruleName>_fragment( <ruleDescriptor.parameterScope:parameterScope(> )
{
<ruleLabelDefs(>
<ruleLabelInitializations(>
<if(trace)>
    ANTLR_PRINTF("enter <ruleName> %d failed = %d, backtracking = %d\\n", this->LT(1),failed,this-
>get_backtracking() );
    <block>
    ANTLR_PRINTF("exit <ruleName> %d, failed = %d, backtracking = %d\\n", this->LT(1),failed,this-
>get_backtracking());

<else>
    <block>
<endif>

goto rule<ruleDescriptor.name>Ex; /* Prevent compiler warnings */
rule<ruleDescriptor.name>Ex; ;
}
// $ANTLR end <ruleName>
>>

synpred(predname) ::= <<

bool <name>::msynpred( antlr3::ClassForwarder< <predname> > )
{
    ANTLR_MARKER start;
    bool success;

    this->inc_backtracking();
    <@start(>
    start = this->mark();
    this->m<predname>_fragment(); // can never throw exception
    success = !( this->get_failedflag() );
    this->rewind(start);
    <@stop(>
    this->dec_backtracking();
    this->set_failedflag(false);
    return success;
}

```



```

    <scopeClean()>
    return <ruleReturnValue()>;
}
<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
/**
 * $ANTLR start <ruleName>
 * <fileName>:<description>
 */
<returnType()>
<name>::<ruleName>(<ruleDescriptor.parameterScope.parameterScope()>)
{
    <if(trace)>ANTLR_PRINTF("enter <ruleName> %s failed=%d, backtracking=%d\n", this->LT(1), this-
>get_backtracking() );<endif>
    <ruleDeclarations()>
    <ruleDescriptor.actions.declarations>
    <ruleLabelDefs()>
    <ruleInitializations()>
    <ruleDescriptor.actions.init>
    <ruleMemoization(rname=ruleName)>
    <ruleLabelInitializations()>

    <if(actions.(actionScope).rulecatch)>
    try {
    <else>
    <if(exceptions)>
        try {
    <endif>
    <endif>
    <@preamble()>
    {
        <block>
    }
    <ruleCleanUp()>

    <if(exceptions)>
    <(ruleDescriptor.actions.after):execAfter()>
    <exceptions:{ e|<catch(decl=e.decl,action=e.action)><n}>}>
    <else>
    <if(!emptyRule)>
    if (this->hasException())
    {
        this->preporterror();

```

```

        this->precover();
        <@setErrorReturnValue()>
    }
    <if(ruleDescriptor.actions.after)>
    else
    {
        <(ruleDescriptor.actions.after):execAfter()>
    }<\n>
    <endif>
<if(actions.(actionScope).rulecatch)>
    } <actions.(actionScope).rulecatch>
    <endif>
<endif>
<endif>
<endif>

    <if(trace)>ANTLR_PRINTF("exit <ruleName> %d failed=%s backtracking=%s\n", this->LT(1), failed, this-
>get_backtracking() );<endif>
    <memoize()>
<if(finally)>
    <finalCode(finalBlock=finally)>
<endif>
    <scopeClean()>
    <@postamble()>
    return <ruleReturnValue()>;
}
/* $ANTLR end <ruleName> */
>>

finalCode(finalBlock) ::= <<
{
    <finalBlock>
}

>>

catch(decl,action) ::= <<
/* catch(decl,action)
*/
}catch (<e.decl>) {
    <e.action>
}
>>

ruleDeclarations() ::= <<

<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType()> retval(this);<\n>
<else>

```

```

<if(PARSER)>
  <name>ImplTraits::RuleReturnValueType _antlr_rule_exit(this);
<endif>
<if(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes: { a |
<a.type> <a.name>;
}>
<endif>
<endif>
<if(memoize)>
ANTLR_MARKER <ruleDescriptor.name>_StartIndex;
<endif>
>>

ruleInitializations() ::= <<
/* Initialize rule variables
*/
<if(ruleDescriptor.returnScope)>
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.returnScope.attributes: { a | <if(a.initValue)>retval.<a.name> = <a.initValue>;<endif> }>
<else>
<ruleDescriptor.returnScope.attributes: { a | <if(a.initValue)><a.name> = <a.initValue>;<endif> }>
<endif>
<endif>
<if(memoize)>
<ruleDescriptor.name>_StartIndex = this->index();<\n>
<endif>
<ruleDescriptor.useScopes: { it | m_<it>_stack.push(<it>Scope()); }; separator="\n">
<ruleDescriptor.ruleScope: { it | m_<it.name>_stack.push(<it.name>Scope()); }; separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: { it | <labelType> <it.label.text> = NULL; }; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
: { it | ImplTraits::TokenPtrsListType list_<it.label.text>; }; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
>>

ruleLabelInitializations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!ruleDescriptor.isSynPred)>
retval.call_start_placeholder();
<endif>
<endif>

```


>>

```
lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | <labelType> <it.label.text> = NULL;}; separator="\n"
>
<ruleDescriptor.charLabels: {it | ANTLR_UINT32 <it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it | ImplTraits::IntTrieType<CommonTokenType>* list_<it.label.text>;}; separator="\n"
>
>>
```

```
lexerRuleLabelInit() ::= <<
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it | list_<it.label.text> = new ImplTraits::IntTrieType<CommonTokenType>(31);}; separator="\n"
>
>>
```

```
lexerRuleLabelFree() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | <it.label.text> = NULL;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it | delete list_<it.label.text>;}; separator="\n"
>
>>
```

```
ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
%>
```

```

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( this->get_backtracking() > 0 ) { this->memoize(<ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }
<endif>
<endif>
>>

```

```

ruleCleanUp() ::= <<

// This is where rules clean up and exit
//
goto rule<ruleDescriptor.name>Ex; /* Prevent compiler warnings */
rule<ruleDescriptor.name>Ex; ;
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
<if(!ruleDescriptor.isSynPred)>
retval.call_stop_placeholder();<\n>
<endif>
<endif>
<endif>
>>

```

```

scopeClean() ::= <<
<ruleDescriptor.useScopes:{it | m_<it>_stack.pop(); }; separator="\n">
<ruleDescriptor.ruleScope:{it | m_<it.name>_stack.pop(); }; separator="\n">

```

```

>>
/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules, which do not produce tokens.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
// Comes from: <block.description>
/** \brief Lexer rule generated by ANTLR3
 *
 * $ANTLR start <ruleName>
 *
 * Looks to match the characters the constitute the token <ruleName>
 * from the attached input stream.
 *
 * \remark
 * - lexer->error == true if an exception was thrown.
 */
void <name>::m<ruleName>(<ruleDescriptor.parameterScope:parameterScope(>))
{
    ANTLR_UINT32 _type;

```

```

<ruleDeclarations()>
<ruleDescriptor.actions.declarations>
<lexerRuleLabelDefs()>
<if(trace)>
std::cout <<< "enter <ruleName> " <<< (char)this->LA(1)
    <<< " line=" <<< this->getLine() <<< ":" <<< this->getCharPositionInLine()
    <<< " failed=" <<< this->get_failedflag() <<< " backtracking=" <<< this->get_backtracking() <<<
std::endl;
<endif>

<if(nakedBlock)>
    <ruleMemoization(rname=ruleName)>
    <lexerRuleLabelInit()>
    <ruleDescriptor.actions.init>

    <block><<\n>
<else>
    <ruleMemoization(rname=ruleName)>
    <lexerRuleLabelInit()>
    _type = <ruleName>;

    <ruleDescriptor.actions.init>

    <block>
this->get_lexstate()->set_type(_type);
<endif>
    <if(trace)>
std::cout <<< "exit <ruleName> " <<< (char)this->LA(1)
    <<< " line=" <<< this->getLine() <<< ":" <<< this->getCharPositionInLine()
    <<< " failed=" <<< this->get_failedflag() <<< " backtracking=" <<< this->get_backtracking() <<<
std::endl;
    <endif>
    <ruleCleanUp()>
    <lexerRuleLabelFree()>
    <(ruleDescriptor.actions.after):execAfter()>
    <memoize>
}
// $ANTLR end <ruleName>
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
/** This is the entry point in to the lexer from an object that
 * wants to generate the next token, such as a pCOMMON_TOKEN_STREAM
 */
void

```

```

<name>::mTokens()
{
    <block><\n>

    goto ruleTokensEx; /* Prevent compiler warnings */
ruleTokensEx: ;
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>
{
    int alt<decisionNumber>=<maxAlt>;
    <decls>
    <@predecision()>
    <decision>
    <@postdecision()>
    <@prebranch()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)> }>
    }
    <@postbranch()>
}
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
{
    // <fileName>:<description>

    ANTLR_UINT32 alt<decisionNumber>;

    alt<decisionNumber>=<maxAlt>;

    <decls>
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)> }>
    }
}

```

>>

```
ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>
```

*/** A special case of a (...) subrule with a single alternative */*

```
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>
```

*/** A (..)+ block with 1 or more alternatives */*

```
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
{
  int cnt<decisionNumber>=0;
  <decls>
  <@preloop()>

  for (;)
  {
    int alt<decisionNumber>=<maxAlt>;
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
      <alts:{ a | <altSwitchCase(i,a)> }>
      default:

      if ( cnt<decisionNumber> >= 1 )
      {
        goto loop<decisionNumber>;
      }
      <ruleBacktrackFailure()>
      <earlyExitEx()>
      <@earlyExitException()>
      goto rule<ruleDescriptor.name>Ex;
    }
  }
}
```

```

cnt<decisionNumber>++;
}
loop<decisionNumber>; /* Jump to here if this rule does not match */
<@postloop()>
}
>>

earlyExitEx() ::= <<
/* mismatchedSetEx()
*/
new ANTLR_Exception<< <name>ImplTraits, EARLY_EXIT_EXCEPTION, StreamType>( this->get_rec(), "" );
<\n>
>>
positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<

// <fileName>:<description>
<decls>

<@preloop()>
for (;;)
{
    int alt<decisionNumber>=<maxAlt>;
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)> }>
    default:
        goto loop<decisionNumber>; /* break out of the loop */
        break;
    }
}
loop<decisionNumber>; /* Jump out to here if this rule does not match */
<@postloop()>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by antlr before code generation
* so we can just use the normal block template
*/
optionalBlock ::= block

```

```
optionalBlockSingleAlt ::= block
```

```
/** A case in a switch that jumps to an alternative given the alternative  
 * number. A DFA predicts the alternative and then a simple switch  
 * does the jump to the code that actually matches that alternative.  
 */
```

```
altSwitchCase(altNum,alt) ::= <<  
case <altNum>:  
  <@prealt()>  
  <alt>  
  break;<\n>  
>>
```

```
/** An alternative is just a list of elements; at outermost level */  
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<  
// <fileName>:<description>  
{  
  <@declarations()>  
  <@initializations()>  
  <elements:element()>  
  <rew>  
  <@cleanup()>  
}  
>>
```

```
// E L E M E N T S
```

```
/** What to emit when there is no rewrite. For auto build  
 * mode, does nothing.  
 */
```

```
noRewrite(rewriteBlockLevel, treeLevel) ::= ""
```

```
/** Dump the elements one per line */  
element(e) ::= <<  
<@prematch()>  
<e.el><\n>  
>>
```

```
/** match a token optionally with a label in front */  
tokenRef(token,label,elementIndex,terminalOptions) ::= <<  
<if(label)><label> = <endif> this->matchToken(<token>, &FOLLOW_<token>_in_<ruleName><elementIndex>);  
<checkRuleBacktrackFailure()>  
>>
```

```
/** ids+=ID */  
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<  
<tokenRef(...)>  
<listLabel(elem=label,...)>  
>>
```

```
listLabel(label,elem) ::= <<
list_<label>.push_back(<elem>);
>>
```

```
/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = this->LA(1);<\n>
<endif>
this->matchc(<char>);
<checkRuleBacktrackFailure()>
>>
```

```
/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = this->LA(1);<\n>
<endif>
this->matchRange(<a>, <b>);
<checkRuleBacktrackFailure()>
>>
```

```
/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,terminalOptions,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label>= this->LA(1);<\n>
<else>
<label>=(<labelType>) this->LT(1);<\n>
<endif>
<endif>
if ( <s> )
{
this->consume();
<postmatchCode>
<if(!LEXER)>
this->set_perror_recovery(false);
<endif>
<if(backtracking)> this->set_failedflag(false); <\n><endif>
}
else
{
<ruleBacktrackFailure()>
<mismatchedSetEx()>
<@mismatchedSetException()>
<if(LEXER)>
```



```

    this->recover();
<else>
<! use following code to make it recover inline;
    this->recoverFromMismatchedSet(&FOLLOW_set_in_<ruleName><elementIndex>);
!>
<endif>
    goto rule<ruleDescriptor.name>Ex;
}<\n>
>>

mismatchedSetEx() ::= <<
new ANTLR_Exception\< <name>ImplTraits, MISMATCHED_SET_EXCEPTION, StreamType>( this->get_rec(),
"" );
<if(PARSER)>
this->get_exception()->set_expectingSet(NULL);
<! use following code to make it recover inline;
this->get_exception()->set_expectingSet( &FOLLOW_set_in_<ruleName><elementIndex> );
!>
<endif>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <<
<if(label)>
ANTLR_MARKER <label>Start = this->getCharIndex();
ANTLR_UINT32 <label>StartLine<elementIndex> = this->getLine();
ANTLR_UINT32 <label>StartCharPos<elementIndex> = this->getCharPositionInLine();
this->matchs(<string>);
<checkRuleBacktrackFailure()>
<label> = new CommonTokenType;
<label>->set_type( CommonTokenType::TOKEN_INVALID );
<label>->set_startIndex( <label>Start);
<label>->set_stopIndex( this->getCharIndex()-1);
<label>->set_input( this->get_input() );
<label>->set_line( <label>StartLine<elementIndex> );
<label>->set_charPositionInLine( <label>StartCharPos<elementIndex> );
<else>
this->matchs(<string>);
<checkRuleBacktrackFailure()><\n>
<endif>
>>

```

```

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>
<label>=<labelType>)this->LT(1);<\n>
<endif>
this->matchAnyToken();
<checkRuleBacktrackFailure()>
>>

```

```

wildcardAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>

```

```

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = this->LA(1);<\n>
<endif>
this->matchAny();
<checkRuleBacktrackFailure()>
>>

```

```

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabel(elem=label,...)>
>>

```

```

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
this->followPush(FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)>m_<scope:delegateName()-><endif><rule.name><(if(args)><args;
separator=", "><endif>);<\n>
this->followPop();
<checkRuleBacktrackFailure()>
>>

```

```

/** ids+=r */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>

```

```

/** A lexer rule reference
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
/* <description> */
<if(label)>
{
  ANTLR_MARKER <label>Start<elementIndex> = this->getCharIndex();
  ANTLR_UINT32 <label>StartLine<elementIndex> = this->getLine();
  ANTLR_UINT32 <label>StartCharPos<elementIndex> = this->getCharPositionInLine();
  <if(scope)>m_<scope:delegateName()-><endif>m<rule.name>(<if(scope)>m_<scope:delegateName()-><endif>
<if(args)>, <endif><args; separator=", ">;
  <checkRuleBacktrackFailure()>
  <label> = new CommonTokenType();
  <label>->set_type( CommonTokenType::TOKEN_INVALID);
  <label>->set_startIndex( <label>Start<elementIndex> );
  <label>->set_stopIndex( this->getCharIndex()-1 );
  <label>->set_input( this->get_input() );
  <label>->set_line( <label>StartLine<elementIndex> );
  <label>->set_charPositionInLine( <label>StartCharPos<elementIndex> );
}
<else>
<if(scope)>m_<scope:delegateName()-><endif>m<rule.name>(<args; separator=", ">;
<checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
{
  ANTLR_UINT32 <label>Start<elementIndex>;
  ANTLR_UINT32 <label>StartLine<elementIndex> = this->getLine();
  ANTLR_UINT32 <label>StartCharPos<elementIndex> = this->getCharPositionInLine();
  <labelType> <label>;
  <label>Start<elementIndex> = this->getCharIndex();
  this->matchc(ANTLR_CHARSTREAM_EOF);
  <checkRuleBacktrackFailure()>
  <label> = new CommonTokenType();
  <label>->set_type( CommonTokenType::TOKEN_EOF );

```

```

<label>->set_startIndex(<label>Start<elementIndex>);
<label>->set_stopIndex(this->getCharIndex()-1);
<label>->set_input( this->get_input() );
<label>->set_line( <label>StartLine<elementIndex> );
<label>->set_charPositionInLine( <label>StartCharPos<elementIndex> );
}
<else>
  this->matchc(ANTLR_CHARSTREAM_EOF);
  <checkRuleBacktrackFailure()>
  <endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg()>"
recRuleArg()            ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> \<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name)  ::= "$<name>=$<src>.<name>;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList, enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( this->LA(1)== CommonTokenType::TOKEN_DOWN ) {
  this->matchToken(CommonTokenType::TOKEN_DOWN, NULL);
  <checkRuleBacktrackFailure()>
  <children:element()>
  this->matchToken(CommonTokenType::TOKEN_UP, NULL);
  <checkRuleBacktrackFailure()>
}
<else>
this->matchToken(CommonTokenType::TOKEN_DOWN, NULL);
<checkRuleBacktrackFailure()>
<children:element()>
this->matchToken(CommonTokenType::TOKEN_UP, NULL);
<checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(...)> )
{
  <ruleBacktrackFailure()>
  <newFPE(...)>

```

```

}
>>

newFPE() ::= <<
ExceptionBaseType* ex = new ANTLR_Exception\< <name>ImplTraits, FAILED_PREDICATE_EXCEPTION,
StreamType>( this->get_rec(), "<description>" );
ex->set_ruleName( "<ruleName>" );
<\n>
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<

{
int LA<decisionNumber>_<stateNumber> = this->LA(<k>);
<edges; separator="\nelse ">
else
{
<if(eotPredictsAlt)>
alt<decisionNumber>=<eotPredictsAlt>;
<else>
<ruleBacktrackFailure()>

<newNVException()>
goto rule<ruleDescriptor.name>Ex;

<endif>
}
}
>>

newNVException() ::= <<
ExceptionBaseType* ex = new ANTLR_Exception\< <name>ImplTraits, NO_VIABLE_ALT_EXCEPTION,
StreamType>( this->get_rec(), "<description>" );
ex->set_decisionNum( <decisionNumber> );
ex->set_state( <stateNumber> );
<@noViableAltException()>
<\n>
>>

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
{

```

```

    int LA<decisionNumber>_<stateNumber> = this->LA(<k>);
    <edges; separator="\nelse ">
}
>>

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */

dfaLoopbackStateDecls() ::= <<
ANTLR_UINT32 LA<decisionNumber>_<stateNumber>;
>>

dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
{
    /* dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState)
    */
    int LA<decisionNumber>_<stateNumber> = this->LA(<k>);
    <edges; separator="\nelse "><n>
    <if(eotPredictsAlt)>
    <if(!edges)>
alt<decisionNumber>=<eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
    else
    {
alt<decisionNumber>=<eotPredictsAlt>;
    }<n>
    <endif>
    <endif>
}
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>";

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( (<labelExpr><if(predicates)> && (<predicates><endif>))
{
    <targetState>
}
>>

```

```

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( this->LA(<k>) )
{
<edges; separator="\n">

default:
<if(eotPredictsAlt)>
  alt<decisionNumber>=<eotPredictsAlt>;
<else>
  <ruleBacktrackFailure()>
  <newNVEException()>
  goto rule<ruleDescriptor.name>Ex;<\n>
<endif>
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( this->LA(<k>) )
{
  <edges; separator="\n">
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( this->LA(<k>) )
{
<edges; separator="\n"><\n>
<if(eotPredictsAlt)>
default:
  alt<decisionNumber>=<eotPredictsAlt>;
  break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels: {it |case <it>:}; separator="\n">
{
  <targetState>
}
  break;
>>

```

```

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = cdfa<decisionNumber>.predict(this, this->get_rec(), this->get_istream(),
cdfa<decisionNumber> );
<checkRuleBacktrackFailure()>
>>

/* Dump DFA tables as static initialized arrays of shorts(16 bits)/characters(8 bits)
 * which are then used to statically initialize the dfa structure, which means that there
 * is no runtime initialization whatsoever, other than anything the C compiler might
 * need to generate. In general the C compiler will lay out memory such that there is no
 * runtime code required.
 */
cyclicDFA(dfa) ::= <<
/** Static dfa state tables for Cyclic dfa:
 * <dfa.description>
 */
static const ANTLR_INT32 dfa<dfa.decisionNumber>_eot[<dfa.numberofStates>] =
{
<dfa.eot; wrap="\n", separator=" ", null="-1">
};
static const ANTLR_INT32 dfa<dfa.decisionNumber>_eof[<dfa.numberofStates>] =
{
<dfa.eof; wrap="\n", separator=" ", null="-1">
};
static const ANTLR_INT32 dfa<dfa.decisionNumber>_min[<dfa.numberofStates>] =
{
<dfa.min; wrap="\n", separator=" ", null="-1">
};
static const ANTLR_INT32 dfa<dfa.decisionNumber>_max[<dfa.numberofStates>] =
{
<dfa.max; wrap="\n", separator=" ", null="-1">
};
static const ANTLR_INT32 dfa<dfa.decisionNumber>_accept[<dfa.numberofStates>] =
{
<dfa.accept; wrap="\n", separator=" ", null="-1">
};
static const ANTLR_INT32 dfa<dfa.decisionNumber>_special[<dfa.numberofStates>] =
{
<dfa.special; wrap="\n", separator=" ", null="-1">
};

/** Used when there is no transition table entry for a particular state */

```



```

static const ANTLR_INT32* dfa<dfa.decisionNumber>_T_empty = NULL;

<dfa.edgeTransitionClassMap.keys:{ table |
static const ANTLR_INT32 dfa<dfa.decisionNumber>_T<i0>[] =
    {
    <table; separator=", ", wrap="\n", null="-1">
    \};<\n>}; null = "">

/* Transition tables are a table of sub tables, with some tables
* reused for efficiency.
*/
static const ANTLR_INT32 * const dfa<dfa.decisionNumber>_transitions[] =
{
    <dfa.transitionEdgeTables:{ xref[dfa<dfa.decisionNumber>_T<xref>}]; separator=", ", wrap="\n", null="NULL">
};

<@errorMethod()>

/* Declare tracking structure for Cyclic DFA <dfa.decisionNumber>
*/
class <name>CyclicDFA<dfa.decisionNumber> : public CyclicDFA\< <name>ImplTraits, <name> >, public
<name>Tokens
{
public:
typedef CyclicDFA\< <name>ImplTraits, <name> > BaseType;
typedef BaseType::ContextType CtxType;

private:
<if(dfa.specialStateSTs)>
//to maintain C-Target compatibility, we need to make some of ctx functions look like member funcs
CtxType* m_ctx;
<endif>

public:
<name>CyclicDFA<dfa.decisionNumber>( ANTLR_INT32 decisionNumber
    , const ANTLR_UCHAR* description
    , const ANTLR_INT32* const eot
    , const ANTLR_INT32* const eof
    , const ANTLR_INT32* const min
    , const ANTLR_INT32* const max
    , const ANTLR_INT32* const accept
    , const ANTLR_INT32* const special
    , const ANTLR_INT32* const *const transition)
:BaseType( decisionNumber, description, eot, eof, min, max, accept,
    special, transition )
{
<if(dfa.specialStateSTs)>
m_ctx = NULL;

```

```

<endif>
}

<if(dfa.specialStateSTs)>
ANTLR_UINT32 LA(ANTLR_INT32 i)
{
    return m_ctx->LA(i);
}

<if(PARSER)>
const CtxType::CommonTokenType* LT(ANTLR_INT32 k)
{
    return m_ctx->LT(k);
}
<endif>
<if(synpreds)>
template<typename PredType>
bool msynpred( PredType pred )
{
    return m_ctx->msynpred(pred);
}
<endif>

ANTLR_INT32 specialStateTransition(CtxType * ctx, RecognizerType* recognizer, IntStreamType* is,
ANTLR_INT32 s)
{
    ANTLR_INT32 _s;

    m_ctx = ctx;
    _s = s;
    switch (s)
    {
        <dfa.specialStateSTs:{state |
        case <i0>:

        <state>}; separator="\n">
    }
    <if(backtracking)>
        if ( ctx->get_backtracking() > 0)
        {
            ctx->set_failedflag( true );
            return -1;
        }
    <endif>
        ExceptionBaseType* ex = new ANTLR_Exception<< <name>ImplTraits, NO_VIABLE_ALT_EXCEPTION,
StreamType>( recognizer, "<dfa.description>" );
        ex->set_decisionNum( <dfa.decisionNumber> );
        ex->set_state(_s);

```

```

    <@noViableAltException()>
    return -1;
}
<endif>
};

static <name>CyclicDFA<dfa.decisionNumber> cdfa<dfa.decisionNumber>(
    <dfa.decisionNumber>, /* Decision number of this dfa */
    /* Which decision this represents: */
    (const ANTLR_UCHAR*)<dfa.description>,
    dfa<dfa.decisionNumber>_eot, /* EOT table */
    dfa<dfa.decisionNumber>_eof, /* EOF table */
    dfa<dfa.decisionNumber>_min, /* Minimum tokens for each state */
    dfa<dfa.decisionNumber>_max, /* Maximum tokens for each state */
    dfa<dfa.decisionNumber>_accept, /* Accept table */
    dfa<dfa.decisionNumber>_special, /* Special transition states */
    dfa<dfa.decisionNumber>_transitions /* Table of transition tables */

);

/* End of Cyclic DFA <dfa.decisionNumber>
* -----
*/
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
* state.
*/
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
{
    ANTLR_UINT32 LA<decisionNumber>_<stateNumber>;<\n>
    ANTLR_MARKER index<decisionNumber>_<stateNumber>;<\n>

    LA<decisionNumber>_<stateNumber> = ctx->LA(1);<\n>
    <if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
    index<decisionNumber>_<stateNumber> = ctx->index();<\n>
    ctx->rewindLast();<\n>
    <endif>
    s = -1;
    <edges; separator="\nelse ">
    <if(semPredState)> <! return input cursor to state before we rewound !>
    ctx->seek(index<decisionNumber>_<stateNumber>);<\n>
    <endif>
    if ( s>=0 )
    {
return s;
    }
}

```

```

}
break;
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( (<labelExpr> <if(predicates)>&& (<predicates>)<endif> )
{
    s = <targetStateNumber>;
}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "( (<left> ) && (<right> ) )"

orPredicates(operands) ::= "(<operands:{o|(<o>)}; separator=\"||\">)"

notPredicate(pred) ::= "!(<evalPredicate(pred,{ })>)"

evalPredicate(pred,description) ::= "(<pred>)"

evalSynPredicate(pred,description) ::= "this->msynpred( antlr3::ClassForwarder\<<pred>>() )"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber> == <atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "this->LA(<k>) == <atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
((LA<decisionNumber>_<stateNumber> >= <lower> ) && (LA<decisionNumber>_<stateNumber> \<= <upper>))
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "(((this->LA(<k>) >=
<lower>) && (this->LA(<k>) \<= <upper>)))"

```

```
setTest(ranges) ::= "<ranges; separator=\\\" || \\\">"
```

```
// A T T R I B U T E S
```

```
makeScopeSet() ::= <<
```

```
/* makeScopeSet()
```

```
*/
```

```
/** Definition of the <scope.name> scope variable tracking
```

```
* structure. An instance of this structure is created by calling
```

```
* <name>_<scope.name>Push().
```

```
*/
```

```
struct <scopeStruct(sname=scope.name,...)>
```

```
{
```

```
/* =====
```

```
* Programmer defined variables...
```

```
*/
```

```
<scope.attributes:{it |<it.decl>;}; separator="\n">
```

```
/* End of programmer defined variables
```

```
* =====
```

```
*/
```

```
};
```

```
>>
```

```
globalAttributeScopeDecl(scope) ::= <<
```

```
<if(scope.attributes)>
```

```
/* globalAttributeScopeDecl(scope)
```

```
*/
```

```
<makeScopeSet(...)>
```

```
<endif>
```

```
>>
```

```
ruleAttributeScopeDecl(scope) ::= <<
```

```
<if(scope.attributes)>
```

```
/* ruleAttributeScopeDecl(scope)
```

```
*/
```

```
<makeScopeSet(...)>
```

```
<endif>
```

```
>>
```

```
globalAttributeScopeDef(scope) ::=
```

```
<<
```

```
/* globalAttributeScopeDef(scope)
```

```
*/
```

```
<if(scope.attributes)>
```

```
StackType\< <scopeStruct(sname=scope.name)> > <scopeStack(sname=scope.name)>;
```

```

<endif>
>>

ruleAttributeScopeDef(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeDef(scope)
*/
StackType\< <scopeStruct(sname=scope.name)> > <scopeStack(sname=scope.name,...)>;

<endif>
>>

scopeStruct(sname) ::= <<
<sname>Scope
>>

scopeStack(sname) ::= <<
m_<sname>_stack
>>

returnStructName(r) ::= "<r.name>_return"

returnType() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.grammar.recognizerName>::<ruleDescriptor:returnStructName()>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
<endif>
<else>
bool
<endif>
%>

/** Generate the C type associated with a single or multiple return
* value(s).
*/
ruleLabelType(referencedRule) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
<referencedRule.name>_return
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>

```

```

<else>
void
<endif>
<endif>
%>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "0".
 */
initValue(typeName) ::= <<
= <cTypeInitMap.(typeName)>
>>

/** Define a rule label */
ruleLabelDef(label) ::= <<
<ruleLabelType(referencedRule=label.referencedRule)> <label.label.text>;
>>
/** Rule label default value */
ruleLabelInitVal(label) ::= <<
>>

ASTLabelType() ::=
"<if(recognizer.ASTLabelType)><recognizer.ASTLabelType><else>ImplTraits::TreeType*<endif>"

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
struct <ruleDescriptor:returnStructName()> : public <name>ImplTraits::RuleReturnValueType
{
public:
typedef <name>ImplTraits::RuleReturnValueType BaseType;
<ruleDescriptor:returnStructName()>()
: BaseType()
<if(scope)>, <scope.attributes:{it | <it.name>() } ; separator=","><endif>
{ init(); }
<ruleDescriptor:returnStructName()>( BaseParserType* parser )
: BaseType(parser)
<if(scope)>, <scope.attributes:{it | <it.name>() } ; separator=","><endif>
{ init(); }

```

```

<ruleDescriptor:returnStructName()>( const <ruleDescriptor:returnStructName()>& other )
    : BaseType(other)
<if(scope)>, <scope.attributes:{it | <it.name>(other.<it.name>) }; separator=", "><endif>
{ copy(other); }
~<ruleDescriptor:returnStructName()>()
{
    <@ruleReturnMembersDelete()>
}

<ruleDescriptor:returnStructName()>&
operator=( const <ruleDescriptor:returnStructName()>& other )
{
    BaseType::operator=( other );
    <if(scope)><scope.attributes:{it | <it.name> = other.<it.name>; }; separator="\n"><endif>
    copy(other);
    return *this;
}
<@ruleReturnMembers()>
void init() { <@ruleReturnMembersInit()> }
void copy( const <ruleDescriptor:returnStructName()>& other ) { <@ruleReturnMembersCopy()> }
<else>
struct <ruleDescriptor:returnStructName()>
{
public:
    <name>ImplTraits::<recognizer.ASTLabelType>    start;
    <name>ImplTraits::<recognizer.ASTLabelType>    stop;
    <ruleDescriptor:returnStructName()>( const <ruleDescriptor:returnStructName()>& other )
    <if(scope.attributes)>
    <scope.attributes:{it | <it.name>(other.<it.name>) }; separator=", ">
    <endif>
    {
        start = other.start;
        stop = other.stop;
    }

    <ruleDescriptor:returnStructName()>&
    operator=( const <ruleDescriptor:returnStructName()>& other )
    {
        start = other.start;
        stop = other.stop;

        <scope.attributes:{it | <it.name> = other.<it.name>; }; separator="\n">
        return *this;
    }
<endif>
    <if(scope)><scope.attributes:{it | <it.type> <it.name>; }; separator="\n"><endif>
};

```



```

<endif>
<endif>
>>

parameterScope(scope) ::= <<
<scope.attributes:{it |<it.decl>} ; separator=", ">
>>

parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name>=<expr>";

/** Note that the scopeAttributeRef does not have access to the
 * grammar name directly
 */
scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
m_<scope>_stack.at( m_<scope>_stack.size()-<negIndex>-1).<attr.name>
<else>
<if(index)>
m_<scope>_stack.at(<index>).<attr.name>
<else>
m_<scope>_stack.peek().<attr.name>
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
m_<scope>_stack.at( m_<scope>_stack.size()-<negIndex>-1).<attr.name> = <expr>;
<else>
<if(index)>
m_<scope>_stack.at(<index>).<attr.name> = <expr>;
<else>
m_<scope>_stack.peek().<attr.name> =<expr>;
<endif>
<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size()>0 && $function::name.equals("foo")}?
 */
isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
<scope>.<attr.name>

```

```

<else>
<scope>
<endif>
>>

returnAttributeRef(ruleDescriptor,attr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
>>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>=<expr>;
<else>
<attr.name>=<expr>;
<endif>
>>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach
//
tokenLabelPropertyRef_text(scope,attr) ::= "<scope>->getText()"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>->get_type()"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>->get_line()"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>->get_charPositionInLine()"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>->get_channel()"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>->get_tokenIndex()"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>->get_tree()"
tokenLabelPropertyRef_int(scope,attr) ::= "<name>ImplTraits::ConvertToInt32(<scope>->getText())"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>.start"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>.stop"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>.tree"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
(this->get_strstream()->toStringSS(<scope>.start, <scope>.start))
<else>
(this->get_strstream()->toStringTT(<scope>.start, <scope>.stop))
<endif>

```

```

>>

ruleLabelPropertyRef_st(scope,attr) ::= "<scope>.st"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "<scope>->get_type()"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "<scope>->get_line()"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "<scope>->get_charPositionInLine()"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "<scope>->get_channel()"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "<scope>->get_tokenIndex()"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "<scope>->getText()"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.start"
rulePropertyRef_stop(scope,attr) ::= "retval.stop"
rulePropertyRef_tree(scope,attr) ::= "retval.tree"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
this->get_input()->toStringSS( this->get_adaptor()->getTokenStartIndex(retval.start), this->get_adaptor()-
>getTokenStopIndex(retval.start))
<else>
this->get_strstream()->toStringTT(retval.start, this->LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "this->getText()"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "this->get_state()->get_tokenStartLine()"
lexerRulePropertyRef_pos(scope,attr) ::= "this->get_state()->get_tokenStartCharPositionInLine()"
lexerRulePropertyRef_channel(scope,attr) ::= "this->get_state()->get_channel()"
lexerRulePropertyRef_start(scope,attr) ::= "this->get_state()->get_tokenStartCharIndex()"
lexerRulePropertyRef_stop(scope,attr) ::= "(this->getCharIndex()-1)"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_int(scope,attr) ::= "<name>ImplTraits::ConvertToInt32(<scope>->getText())"

// setting $st and $tree is allowed in local rule. everything else is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree=<expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st=<expr>;"

/** How to deal with an @after for C targets. Because we cannot rely on
 * any garbage collection, after code is executed even in backtracking
 * mode. Must be documented clearly.
 */

```

```

execAfter(action) ::= <<
{
  <action>
}
>>

/** How to execute an action (when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>
<if(actions.(actionScope).synpredgate)>
if ( <actions.(actionScope).synpredgate> )
{
  <action>
}
<else>
if ( BACKTRACKING == 0 )
{
  <action>
}
<endif>
<else>
{
  <action>
}
<endif>
>>

// M I S C (properties, etc...)

bitsetDeclare(bitsetname, words64, traits) ::= <<

/** Bitset defining follow set for error recovery in rule state: <name> */
static ANTLR_BITWORD <bitsetname>_bits[] = { <words64:{it |ANTLR_UINT64_LIT(<it>)}; separator=", "> };
static <traits>::BitsetListType <bitsetname>(<bitsetname>_bits, <length(words64)>);
>>

codeFileExtension() ::= ".cpp"

true_value() ::= "true"
false_value() ::= "false"

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Cpp/Cpp.stg
No license file was found, but licenses were detected in source scan.

/*
[The "BSD license"]
Copyright (c) 2005-2006 Terence Parr

```

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to AST stuff. Dynamic inheritance

* hierarchy is set up as ASTDbg : AST : Dbg : Java by code generator.

*/

```
parserMembers() ::= <<
protected DebugTreeAdaptor adaptor;
public void setTreeAdaptor(TreeAdaptor adaptor) {
<if(grammar.grammarIsRoot)>
    this.adaptor = new DebugTreeAdaptor(dbg,adaptor);
<else>
    this.adaptor = (DebugTreeAdaptor)adaptor; // delegator sends dbg adaptor
<endif>
    <grammar.directDelegates:{g|<g:delegateName()>.setTreeAdaptor(this.adaptor);}>
    }
    public TreeAdaptor getTreeAdaptor() {
        return adaptor;
    }
}>>
```

```
parserCtorBody() ::= <<
<super.parserCtorBody()>
>>
```

```

createListenerAndHandshake() ::= <<
DebugEventSocketProxy proxy =
  new DebugEventSocketProxy(this,port,<if(TREE_PARSER)>input.getTreeAdaptor()<else>adaptor<endif>);
setDebugListener(proxy);
set<inputStreamType>(new Debug<inputStreamType>(input,proxy));
try {
  proxy.handshake();
}
catch (IOException ioe) {
  reportError(ioe);
}
>>

```

```

@ctorForRootGrammar.finally() ::= <<
TreeAdaptor adap = new CommonTreeAdaptor();
setTreeAdaptor(adap);
proxy.setTreeAdaptor(adap);
>>

```

```

@ctorForProfilingRootGrammar.finally() ::= <<
TreeAdaptor adap = new CommonTreeAdaptor();
setTreeAdaptor(adap);
>>

```

```

@ctorForPredefinedListener.superClassRef() ::= "super(input, dbg);"

```

```

@ctorForPredefinedListener.finally() ::= <<
<if(grammar.grammarIsRoot)> <! don't create new adaptor for delegates !>
TreeAdaptor adap = new CommonTreeAdaptor();
setTreeAdaptor(adap);
<endif>
>>

```

```

@rewriteElement.pregen() ::= "dbg.location(<e.line>,<e.pos>);"

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Java/ASTDbg.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

```

[The "BSD license"]

```

```

Copyright (c) 2005-2009 Terence Parr

```

```

All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Template overrides to add debugging to AST stuff. Dynamic inheritance

* hierarchy is set up as ASTDbg : AST : Dbg : Python by code generator.

*/

group ASTDbg;

astAccessor() ::= <<

def setTreeAdaptor(self, adaptor):

<if(grammar.grammarIsRoot)>

self._adaptor = DebugTreeAdaptor(self.dbg, adaptor)

<else>

self._adaptor = adaptor # delegator sends dbg adaptor

<endif>

<grammar.directDelegates:{g|<g.delegateName()>.setTreeAdaptor(self._adaptor)}>

def getTreeAdaptor(self):

return self._adaptor

adaptor = property(getTreeAdaptor, setTreeAdaptor)<\n>

>>

createListenerAndHandshake() ::= <<

proxy = DebugEventSocketProxy(self,

adaptor=<if(TREE_PARSER)>self.input.getTreeAdaptor(<else>self._adaptor<endif>,

debug=debug_socket, port=port)

self.setDebugListener(proxy)

self.adaptor.setDebugListener(proxy)

self.input.setDebugListener(proxy)

#self.set<inputStreamType>(Debug<inputStreamType>(self.input, proxy))

```
proxy.handshake()
```

```
>>
```

```
@rewriteElement.pregen() ::= "self._dbg.location(<e.line>, <e.pos>)"
```

```
Found in path(s):
```

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/Python/ASTDbg.stg
```

```
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
<if(!TREE_PARSER)><! tree parser would already have imported !>
```

```
import org.antlr.runtime.tree.*;<\n>
```

```
<endif>
```

```
>>
```

```
@genericParser.members() ::= <<
```

```
<@super.members(>
```

```
<parserMembers(>
```


>>

```
/** Add an adaptor property that knows how to build trees */  
parserMembers() ::= <<  
protected TreeAdaptor adaptor = new CommonTreeAdaptor();
```

```
public void setTreeAdaptor(TreeAdaptor adaptor) {  
    this.adaptor = adaptor;  
    <grammar.directDelegates: {g|<g:delegateName().setTreeAdaptor(this.adaptor);}>  
}  
public TreeAdaptor getTreeAdaptor() {  
    return adaptor;  
}  
>>
```

```
@returnScope.ruleReturnMembers() ::= <<  
<ASTLabelType> tree;  
@Override  
public <ASTLabelType> getTree() { return tree; }  
>>
```

```
/** Add a variable to track rule's return AST */  
ruleDeclarations() ::= <<  
<super.ruleDeclarations()>  
<ASTLabelType> root_0 = null;<\n>  
>>
```

```
ruleLabelDefs() ::= <<  
<super.ruleLabelDefs()>  
<if(!ruleDescriptor.isSynPred)>  
<[ruleDescriptor.tokenLabels,ruleDescriptor.wildcardTreeLabels,  
    ruleDescriptor.wildcardTreeListLabels]:{it | <ASTLabelType> <it.label.text>_tree=null;}; separator="\n">  
<ruleDescriptor.tokenListLabels:{it | <ASTLabelType> <it.label.text>_tree=null;}; separator="\n">  
<if(ruleDescriptor.supportsLabelOptimization)>  
<ruleDescriptor.allTokenRefsInRewrites  
    :{it | RewriteRule<rewriteElementType>Stream stream_<it>=new  
    RewriteRule<rewriteElementType>Stream(adaptor,"token <it>");}; separator="\n">  
<ruleDescriptor.allRuleRefsInRewrites  
    :{it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule <it>");};  
    separator="\n">  
<else>  
<ruleDescriptor.allTokenRefsInAltsWithRewrites  
    :{it | RewriteRule<rewriteElementType>Stream stream_<it>=new  
    RewriteRule<rewriteElementType>Stream(adaptor,"token <it>");}; separator="\n">  
<ruleDescriptor.allRuleRefsInAltsWithRewrites  
    :{it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule <it>");};  
    separator="\n">  
<endif>
```

```

<endif>
>>

/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode && !ruleDescriptor.isSynPred)>
root_0 = (<ASTLabelType>)adaptor.nil();<\n>
<endif>
<endif>
<endif>
>>

// Tracking Rule Elements

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
 * to the tracking list stream_ID for use in the rewrite.
 */
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label, ...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label, ...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<

```

```

<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule.name>.add(<label>.getTree());
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabel(label, {<label>.getTree()})>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule>.add(<label>.getTree());
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabel(label, {<label>.getTree()})>
>>

// R e w r i t e

rewriteCode(
alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
referencedWildcardLabels,
referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::=
<<

// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
// wildcard labels: <referencedWildcardLabels,referencedWildcardListLabels; separator=", ">
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
<endif>
<prevRuleRootRef(>.tree = root_0;
<rewriteCodeLabels(>

```

```

root_0 = (<ASTLabelType>)adaptor.nil();
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER)>
<if(rewriteMode)>
<prevRuleRootRef(>.tree = (<ASTLabelType>)adaptor.rulePostProcessing(root_0);
input.replaceChildren(adaptor.getParent(retval.start),
    adaptor.getChildIndex(retval.start),
    adaptor.getChildIndex(_last),
    retval.tree);
<endif>
<endif>
<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER)>
<prevRuleRootRef(>.tree = root_0;
<else>
<if(!rewriteMode)>
<prevRuleRootRef(>.tree = root_0;
<endif>
<endif>
<if(backtracking)>
}
<endif>
>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
:{it | RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it>",<it>)};
separator="\n"
>
<referencedTokenListLabels
:{it | RewriteRule<rewriteElementType>Stream stream_<it>=new
RewriteRule<rewriteElementType>Stream(adaptor,"token <it> ", list_<it>)};
separator="\n"
>
<referencedWildcardLabels
:{it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedWildcardListLabels
:{it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"wildcard <it> ",list_<it>)};
separator="\n"
>
<referencedRuleLabels
:{it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"rule
<it>",<it>!=null?<it>.getTree():null)};
separator="\n"

```

```

>
<referencedRuleListLabels
: {it | RewriteRuleSubtreeStream stream_<it>=new RewriteRuleSubtreeStream(adaptor,"token <it>","list_<it>");
separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
if ( <referencedElementsDeep:{el | stream_<el>.hasNext()}; separator="||"> ) {
<alt>
}
<referencedElementsDeep:{el | stream_<el>.reset();<\n>}>
>>

rewriteClosureBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
while ( <referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) {
<alt>
}
<referencedElements:{el | stream_<el>.reset();<\n>}>
>>

rewritePositiveClosureBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
if ( !(<referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) ) {
throw new RewriteEarlyExitException();
}
while ( <referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) {
<alt>
}
}

```

```

<referencedElements:{el | stream_<el>.reset();<\n}>
>>

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>) {
  <a.alt>
}<\n>
<else>
{
  <a.alt>
}<\n>
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
<ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.nil();
<root:rewriteElement()>
<children:rewriteElement()>
adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,args,terminalOptions={}) ::= <<
adaptor.addChild(root_<treeLevel>, <createRewriteNodeFromElement(...)>);
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<

```

```

adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,args,terminalOptions={}) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<createRewriteNodeFromElement(...)>,
root_<treeLevel>);
>>

rewriteImaginaryTokenRef(args,token,elementIndex,terminalOptions={}) ::= <<
adaptor.addChild(root_<treeLevel>, <createImaginaryNode(token,args,terminalOptions)>);
>>

rewriteImaginaryTokenRefRoot(args,token,elementIndex,terminalOptions={}) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<createImaginaryNode(token,args,terminalOptions)>,
root_<treeLevel>);
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
adaptor.addChild(root_<treeLevel>, stream_<rule>.nextTree());
>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(stream_<rule>.nextNode(), root_<treeLevel>);
>>

rewriteNodeAction(action) ::= <<
adaptor.addChild(root_<treeLevel>, <action>);

```

>>

```
rewriteNodeActionRoot(action) ::= <<  
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<action>, root_<treeLevel>);  
>>
```

```
/** Gen $ruleLabel ... where defined via ruleLabel=rule */  
rewriteRuleLabelRef(label) ::= <<  
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());  
>>
```

```
/** Gen $ruleLabel ... where defined via ruleLabel+=rule */  
rewriteRuleListLabelRef(label) ::= <<  
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());  
>>
```

```
/** Gen ^($ruleLabel ...) where ruleLabel=rule */  
rewriteRuleLabelRefRoot(label) ::= <<  
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);  
>>
```

```
/** Gen ^($ruleLabel ...) where ruleLabel+=rule */  
rewriteRuleListLabelRefRoot(label) ::= <<  
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);  
>>
```

```
rewriteWildcardLabelRef(label) ::= <<  
adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());  
>>
```

```
createImaginaryNode(tokenType,args,terminalOptions={ }) ::= <%  
<if(terminalOptions.node)>  
<! new MethodNode(IDLabel, args) !>  
new <terminalOptions.node>(<tokenType><if(args)>, <args; separator=", "><endif>)  
<else>  
(<ASTLabelType>)adaptor.create(<tokenType>, <args; separator=", "><if(!args)>"<tokenType>"<endif>)  
<endif>  
>%
```

```
createRewriteNodeFromElement(token,args,terminalOptions={ }) ::= <%  
<if(terminalOptions.node)>  
new <terminalOptions.node>(stream_<token>.nextToken(<if(args)>, <args; separator=", "><endif>)  
<else>  
<if(args)> <! must create new node from old !>  
adaptor.create(<token>, <args; separator=", ">)  
<else>  
stream_<token>.nextNode()
```



```
<endif>
<endif>
%>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Java/AST.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2012 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Template subgroup to add template rewrite output
```

```
* If debugging, then you'll also get STDbg.stg loaded.
```

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
import stringtemplate3
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturnInit() ::= <<
```

```
self.st = None
```

```
>>
```

```

@returnScope.ruleReturnMembers() ::= <<
def getTemplate(self):
    return self.st

def toString(self):
    if self.st is not None:
        return self.st.toString()
    return None
__str__ = toString

>>

@genericParser.init() ::= <<
<@super.init()>
self.templateLib = stringtemplate3.StringTemplateGroup(
    '<name>Templates', lexer='angle-bracket'
)

>>

@genericParser.members() ::= <<
<@super.members()>
def setTemplateLib(self, templateLib):
    self.templateLib = templateLib

def getTemplateLib(self):
    return self.templateLib

>>

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.st})>
>>

rewriteTemplate(alts) ::= <<
# TEMPLATE REWRITE
<if(backtracking)>
if <actions.(actionScope).synpredgate>:
    <first(alts):rewriteTemplateAltFirst()>
    <rest(alts):{it | el<rewriteTemplateAlt(it)>}>
    <if(rewriteMode)><replaceTextInLine()><endif>

<else>
<first(alts):rewriteTemplateAltFirst()>
<rest(alts):{it | el<rewriteTemplateAlt(it)>}>

```

```
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>
```

```
replaceTextInLine() ::= <<
<if(TREE_PARSER)>
self.input.getTokenStream().replace(
  self.input.getTreeAdaptor().getTokenStartIndex(retval.start),
  self.input.getTreeAdaptor().getTokenStopIndex(retval.start),
  retval.st
)
<else>
self.input.replace(
  retval.start.getTokenIndex(),
  self.input.LT(-1).getTokenIndex(),
  retval.st
)
<endif>
>>
```

```
rewriteTemplateAltFirst(alt) ::= <<
<if(alt.pred)>
if <alt.pred>:
  # <alt.description>
  retval.st = <alt.alt>
<\n>
<else>
# <alt.description>
retval.st = <alt.alt>
<\n>
<endif>
>>
```

```
rewriteTemplateAlt(alt) ::= <<
<if(alt.pred)>if <alt.pred>:
  # <alt.description>
  retval.st = <alt.alt>
<\n>
<else>se:
  # <alt.description>
  retval.st = <alt.alt>
<\n>
<endif>
>>
```

```
rewriteEmptyTemplate(alts) ::= <<
None
>>
```

```

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template after having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded
 * template.
 */
rewriteExternalTemplate(name,args) ::= <%
self.templateLib.getInstanceOf("<name><if(args)>, attributes={<args:{a | "<a.name>": <a.value>}; separator=",
"><endif>")
%>

```

```

/** expr is a string expression that says what template to load */
rewriteIndirectTemplate(expr,args) ::= <%
self.templateLib.getInstanceOf(<expr><if(args)>, attributes={<args:{a | "<a.name>": <a.value>}; separator=",
"><endif>")
%>

```

```

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <%
stringtemplate3.StringTemplate("<template>", group=self.templateLib<if(args)>, attributes={<args:{a |
"<a.name>": <a.value>}; separator=", "><endif>")
%>

```

```

/** plain -> {foo} action */
rewriteAction(action) ::= <<
<action>
>>

```

```

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
(<st>)[("<attrName>")] = <expr>
>>

```

```

/** Translate % {stringExpr} */
actionStringConstructor(stringExpr) ::= <<
stringtemplate3.StringTemplate(<stringExpr>, group=self.templateLib)
>>

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python3/ST.stg

```

No license file was found, but licenses were detected in source scan.

```

/*

```

```

[The "BSD license"]

```

```

Copyright (c) 2005-2006 Terence Parr

```

```

All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Template subgroup to add template rewrite output
```

```
* If debugging, then you'll also get STDbg.stg loaded.
```

```
*/
```

```
@outputFile.imports() ::= <<
```

```
<@super.imports(>
```

```
import org.antlr.stringtemplate.*;
```

```
import org.antlr.stringtemplate.language.*;
```

```
import java.util.HashMap;
```

```
>>
```

```
/** Add this to each rule's return value struct */
```

```
@returnScope.ruleReturnMembers() ::= <<
```

```
public StringTemplate st;
```

```
public Object getTemplate() { return st; }
```

```
public String toString() { return st==null?null:st.toString(); }
```

```
>>
```

```
@genericParser.members() ::= <<
```

```
<@super.members(>
```

```
protected StringTemplateGroup templateLib =
```

```
new StringTemplateGroup("<name>Templates", AngleBracketTemplateLexer.class);
```

```
public void setTemplateLib(StringTemplateGroup templateLib) {
```

```
this.templateLib = templateLib;
```

```

}
public StringTemplateGroup getTemplateLib() {
    return templateLib;
}
/** allows convenient multi-value initialization:
 * "new STAttrMap().put(...).put(...)"
 */
@SuppressWarnings("serial")
public static class STAttrMap extends HashMap<String, Object> {
    public STAttrMap put(String attrName, Object value) {
        super.put(attrName, value);
        return this;
    }
}
>>

```

```

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, {<label>.getTemplate()})>
>>

```

```

rewriteTemplate(alts) ::= <<

```

```

// TEMPLATE REWRITE
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {
    <alts:rewriteTemplateAlt(); separator="else ">
    <if(rewriteMode)><replaceTextInLine()><endif>
}
<else>
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>

```

```

replaceTextInLine() ::= <<
<if(TREE_PARSER)>
((TokenRewriteStream)input.getTokenStream()).replace(
    input.getTreeAdaptor().getTokenStartIndex(retval.start),
    input.getTreeAdaptor().getTokenStopIndex(retval.start),
    retval.st);
<else>
((TokenRewriteStream)input).replace(
    ((Token)retval.start).getTokenIndex(),
    input.LT(-1).getTokenIndex(),
    retval.st);
<endif>

```

```
>>
```

```
rewriteTemplateAlt(alt) ::= <<  
// <alt.description>  
<if(alt.pred)>  
<if(alt.pred)> {  
  retval.st = <alt.alt>;  
}<\n>  
<else>  
{  
  retval.st = <alt.alt>;  
}<\n>  
<endif>  
>>
```

```
rewriteEmptyTemplate(alts) ::= <<  
null;  
>>
```

```
/** Invoke a template with a set of attribute name/value pairs.  
 * Set the value of the rule's template after having set  
 * the attributes because the rule's template might be used as  
 * an attribute to build a bigger template; you get a self-embedded  
 * template.  
 */
```

```
rewriteExternalTemplate(name,args) ::= <%  
templateLib.getInstanceOf("<name>"<if(args)>,  
  new STAttrMap()<args:{a | .put("<a.name>", <a.value>)}>  
<endif>)  
>%
```

```
/** expr is a string expression that says what template to load */  
rewriteIndirectTemplate(expr,args) ::= <%  
templateLib.getInstanceOf(<expr><if(args)>,  
  new STAttrMap()<args:{a | .put("<a.name>", <a.value>)}>  
<endif>)  
>%
```

```
/** Invoke an inline template with a set of attribute name/value pairs */  
rewriteInlineTemplate(args, template) ::= <%  
new StringTemplate(templateLib, "<template>"<if(args)>,  
  new STAttrMap()<args:{a | .put("<a.name>", <a.value>)}>  
<endif>)  
>%
```

```
/** plain -> {foo} action */  
rewriteAction(action) ::= <<  
<action>
```

>>

```
/** An action has %st.attrName=expr; or % {st}.attrName=expr; */  
actionSetAttribute(st,attrName,expr) ::= <<  
(<st>).setAttribute("<attrName>",<expr>);  
>>
```

```
/** Translate % {stringExpr} */  
actionStringConstructor(stringExpr) ::= <<  
new StringTemplate(templateLib,<stringExpr>)  
>>
```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Java/ST.stg
No license file was found, but licenses were detected in source scan.

```
/*  
[The "BSD license"]  
Copyright (c) 2005-2006 Terence Parr  
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Templates for building ASTs during tree parsing.  
*  
* Deal with many combinations. Dimensions are:  
* Auto build or rewrite
```



```

* no label, label, list label (label/no-label handled together)
* child, root
* token, set, rule, wildcard
*
* Each combination has its own template except that label/no label
* is combined into tokenRef, ruleRef, ...
*/

/** Add a variable to track last element matched */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
var _first_0:<ASTLabelType> = null;
var _last:<ASTLabelType> = null;<\n>
>>

/** What to emit when there is no rewrite rule. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= <<
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(rewriteMode)>
retval.tree = <ASTLabelType>(_first_0);
if ( adaptor.getParent(retval.tree)!=null && adaptor.isNil( adaptor.getParent(retval.tree) ) )
    retval.tree = <ASTLabelType>(adaptor.getParent(retval.tree));
<endif>
<if(backtracking)>}<endif>
>>

/** match ^(root children) in tree parser; override here to
* add tree construction actions.
*/
tree(root, actionsAfterRoot, children, nullableChildList,
    enclosingTreeLevel, treeLevel) ::= <<
_last = <ASTLabelType>(input.LT(1));
{
var _save_last_<treeLevel>:<ASTLabelType> = _last;
var _first_<treeLevel>:<ASTLabelType> = null;
<if(!rewriteMode)>
var root_<treeLevel>:<ASTLabelType> = <ASTLabelType>(adaptor.nil());
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
<if(root.el.rule)>
if ( _first_<enclosingTreeLevel>==null ) _first_<enclosingTreeLevel> = <root.el.label>.tree;
<else>
if ( _first_<enclosingTreeLevel>==null ) _first_<enclosingTreeLevel> = <root.el.label>;
<endif>
}
}

```

```

<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==TokenConstants.DOWN ) {
    matchStream(input, TokenConstants.DOWN, null); <checkRuleBacktrackFailure()>
    <children:element()>
    matchStream(input, TokenConstants.UP, null); <checkRuleBacktrackFailure()>
}
<else>
matchStream(input, TokenConstants.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
matchStream(input, TokenConstants.UP, null); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}<\n>
>>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.tokenRef(...)>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = <ASTLabelType>(adaptor.dupNode(<label>));
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
>>

```

```

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = <ASTLabelType>(adaptor.dupNode(<label>));
<endif><\n>
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(<label>_tree, root_<treeLevel>));
<if(backtracking)>}<endif>
<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.wildcard(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.dupTree(<label>);
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
>>

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_last = <ASTLabelType>(input.LT(1));

```

```

<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = <ASTLabelType>(adaptor.dupNode(<label>));
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
>>

matchSetBang(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.matchSet(...)>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = <ASTLabelType>(adaptor.dupNode(<label>));
<endif><\n>
root_<treeLevel> = <ASTLabelType>(adaptor.becomeRoot(<label>_tree, root_<treeLevel>));
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
>>

// RULE REF AST

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>

```

```

<if(!rewriteMode)>
adaptor.addChild(root_<treeLevel>, <label>.tree);
<else> <! rewrite mode !>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>.tree;
<endif>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.tree })>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>root_<treeLevel> =
<ASTLabelType>(adaptor.becomeRoot(<label>.tree, root_<treeLevel>));
<endif>
>>

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(label, {<label>.tree })>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.ruleRefTrackAndListLabel(...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
_last = <ASTLabelType>(input.LT(1));
<super.ruleRefRootTrack(...)>
>>

/** ^(x+=rule ...) rewrite */

```

```
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  _last = <ASTLabelType>(input.LT(1));
  <super.ruleRefRuleRootTrackAndListLabel(...)>
>>
```

```
/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
```

```
createRewriteNodeFromElement(token,terminalOptions,scope) ::= <<
  <if(terminalOptions.node)>
  new <terminalOptions.node>(stream_<token>.nextNode())
  <else>
  stream_<token>.nextNode()
  <endif>
>>
```

```
ruleCleanup() ::= <<
  <super.ruleCleanup()>
  <if(!rewriteMode)>
  <if(backtracking)>if ( <actions.(actionScope).synpredgate> ) {<\n><endif>
  retval.tree = <ASTLabelType>(adaptor.rulePostProcessing(root_0));
  <if(backtracking)>}<endif>
  <endif>
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/ActionScript/ASTTreeParser.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
Copyright (c) 2008 Erik van Bilzen
Copyright (c) 2007-2008 Johannes Luber
Copyright (c) 2005-2007 Kunle Odutola
Copyright (c) 2005-2006 Terence Parr
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

group ASTTreeParser;

/** Add a variable to track last element matched */

ruleDeclarations() ::= <<

<super.ruleDeclarations()>

_First[0] := nil;

_Last := nil;<\n>

>>

ruleDeclarationVars() ::= <<

<super.ruleDeclarationVars()>

_First, _Save_Last: array [0..63] of I<ASTLabelType>;

_Last: I<ASTLabelType>;

>>

/** What to emit when there is no rewrite rule. For auto build

* mode, does nothing.

*/

noRewrite(rewriteBlockLevel, treeLevel) ::= <<

<if(backtracking)>if (State.Backtracking = 0) then

begin<endif>

<if(rewriteMode)>

RetVal.Tree := _First[0] as I<ASTLabelType>;

if (Adaptor.GetParent(RetVal.Tree) \<\> nil) and (Adaptor.IsNil(Adaptor.GetParent(RetVal.Tree))) then

```

RetVal.Tree := Adaptor.GetParent(RetVal.Tree) as I<ASTLabelType>;
<endif>
<if(backtracking)>end;<endif>
>>

/** match ^(root children) in tree parser; override here to
 * add tree construction actions.
 */
tree(root, actionsAfterRoot, children, nullableChildList,
      enclosingTreeLevel, treeLevel) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
begin
_Save_Last[<treeLevel>] := _Last;
_First[<treeLevel>] := nil;
<if(!rewriteMode)>
Root[<treeLevel>] := Adaptor.GetNilNode as I<ASTLabelType>;<\n>
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>
<if(root.el.rule)>
if (_First[<enclosingTreeLevel>] = nil) then _First[<enclosingTreeLevel>] := <root.el.label>.Tree;
<else>
if (_First[<enclosingTreeLevel>] = nil) then _First[<enclosingTreeLevel>] := <root.el.label>;
<endif>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if (Input.LA(1) = TToken.DOWN) then
begin
Match(Input, TToken.DOWN, nil); <checkRuleBacktrackFailure()>
<children:element()>
Match(Input, TToken.UP, nil); <checkRuleBacktrackFailure()>
end;
<else>
Match(Input, TToken.DOWN, nil); <checkRuleBacktrackFailure()>
<children:element()>
Match(Input, TToken.UP, nil); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
Adaptor.AddChild(Root[<enclosingTreeLevel>], Root[<treeLevel>]);
<endif>
_Last := _Save_Last[<treeLevel>];
end;<\n>
>>

// TOKEN AST STUFF

```



```

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.tokenRef(...)>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin<\n>
<endif>
<if(terminalOptions.node)>
<label>_tree := T<terminalOptions.node>.Create(<label>);
<else>
<label>_tree := Adaptor.DupNode(<label>) as I<ASTLabelType>;
<endif><\n>
Adaptor.AddChild(Root[<treeLevel>], <label>_tree);
<if(backtracking)>
end;
<endif>
<else> <! rewrite mode !>
<if(backtracking)>if (State.Backtracking = 0) then <endif>
if (_First[<treeLevel>] = nil) then _First[<treeLevel>] := <label>;
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>
if (State.Backtracking = 0) then
begin
<endif>
<if(terminalOptions.node)>

```

```

<label>_tree := T<terminalOptions.node>.Create(<label>);
<else>
<label>_tree := Adaptor.DupNode(<label>) as I<ASTLabelType>;
<endif><\n>
Root[<treeLevel>] := Adaptor.BecomeRoot(<label>_tree, Root[<treeLevel>]) as I<ASTLabelType>;
<if(backtracking)>
end;
<endif>
<endif>
>>

```

```

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

```

```

// SET AST

```

```

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_Llast := Input.LT(1) as I<ASTLabelType>;
<super.matchSet(..., postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if (State.Backtracking = 0) then begin <endif>
<if(terminalOptions.node)>
<label>_tree := T<terminalOptions.node>.Create(<label>);
<else>
<label>_tree := Adaptor.DupNode(<label>) as I<ASTLabelType>;
<endif><\n>
Adaptor.AddChild(Root[<treeLevel>], <label>_tree);
<if(backtracking)>end;<endif>
<endif>
}
)>
>>

```

```

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
<noRewrite()> <! set return tree !>
>>

```

```

matchSetBang(s,label,elementIndex,postmatchCode) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.matchSet(...)>
>>

```

```

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<super.matchSet(..., postmatchCode={

```

```

<if(!rewriteMode)>
<if(backtracking)>if (State.Backtracking = 0) then begin <endif>
<if(terminalOptions.node)>
<label>_tree := T<terminalOptions.node>.Create(<label>);
<else>
<label>_tree := Adaptor.DupNode(<label>) as I<ASTLabelType>;
<endif><\n>
Root[<treeLevel>] := Adaptor.BecomeRoot(<label>_tree, Root[<treeLevel>]) as I<ASTLabelType>;
<if(backtracking)>end;<endif>
<endif>
}
)>
>>

```

```
// RULE REF AST
```

```

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.ruleRef(...)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>
<if(!rewriteMode)>
Adaptor.AddChild(Root[<treeLevel>], <label>.Tree);
<else> <! rewrite mode !>
if (_First[<treeLevel>] = nil) then _First[<treeLevel>] := <label>.Tree;
<endif>
>>

```

```

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label+".Tree",...)>
>>

```

```

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
_Last := Input.LT(1) as I<ASTLabelType>;
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (State.Backtracking = 0) then <endif>Root[<treeLevel>] :=
Adaptor.BecomeRoot(<label>.Tree, Root[<treeLevel>]) as I<ASTLabelType>;
<endif>
>>

```

```

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(elem=label+".Tree",...)>

```

```

>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
  _Last := Input.LT(1) as I<ASTLabelType>;
  <super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  _Last := Input.LT(1) as I<ASTLabelType>;
  <super.ruleRefTrackAndListLabel(...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
  _Last := Input.LT(1) as I<ASTLabelType>;
  <super.ruleRefRuleRootTrack(...)>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  _Last := Input.LT(1) as I<ASTLabelType>;
  <super.ruleRefRuleRootTrackAndListLabel(...)>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,terminalOptions,scope) ::= <<
  <if(terminalOptions.node)>
  T<terminalOptions.node>.Create((Locals['Stream_<token>'] as IRewriteRuleElementStream).NextNode)
  <else>
  (Locals['Stream_<token>'] as IRewriteRuleElementStream).NextNode
  <endif>
>>

ruleCleanup() ::= <<
  <super.ruleCleanup()>
  <if(!rewriteMode)>
  <if(backtracking)>
  if (State.Backtracking = 0) then
  begin<\n>
  <endif>
  RetVal.Tree := Adaptor.RulePostProcessing(Root[0]) as I<ASTLabelType>;
  <if(backtracking)>
  end;
  <endif>

```

<endif>

>>

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Delphi/ASTTreeParser.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2011 Terence Parr

* All rights reserved.

*

* Conversion to C#:

* Copyright (c) 2011 Sam Harwell, Pixel Mine, Inc.

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label
 * is combined into tokenRef, ruleRef, ...
 */

```
/** Add a variable to track last element matched */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
<if(!ruleDescriptor.isSynPred)>
<ASTLabelType> _first_0 = default(<ASTLabelType>);
<ASTLabelType> _last = default(<ASTLabelType>);
<endif>
>>
```

```
/** What to emit when there is no rewrite rule. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel=false, treeLevel=false) ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(rewriteMode)>
retval.Tree = (<ASTLabelType>)_first_0;
if (adaptor.GetParent(retval.Tree)!=null && adaptor.IsNil(adaptor.GetParent(retval.Tree)))
retval.Tree = (<ASTLabelType>)adaptor.GetParent(retval.Tree);
<endif>
<if(backtracking)>}<endif>
<endif>
>>
```

```
/** match ^(root children) in tree parser; override here to
* add tree construction actions.
*/
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
{
<ASTLabelType> _save_last_<treeLevel> = _last;
<ASTLabelType> _first_<treeLevel> = default(<ASTLabelType>);
<if(!rewriteMode)>
<ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.Nil();
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
<if(root.el.rule)>
if (_first_<enclosingTreeLevel> == null) _first_<enclosingTreeLevel> = <root.el.label>.Tree;
<else>
if (_first_<enclosingTreeLevel> == null) _first_<enclosingTreeLevel> = <root.el.label>;
```

```

<endif>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if (input.LA(1) == TokenType.Down) {
    Match(input, TokenType.Down, null); <checkRuleBacktrackFailure()>
    <children:element()>
    Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
}
<else>
Match(input, TokenType.Down, null); <checkRuleBacktrackFailure()>
<children:element()>
Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
adaptor.AddChild(root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}
<else>
<super.tree(...)>
<endif>
>>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<else>
<super.tokenRefBang(...)>
<endif>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.tex

```

```

t)>,<terminalOptions.text; format="string"><endif>;
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.text)>,<terminalOptions.text; format="string"><endif>;
<endif><\n>
adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>;
<endif>
<else>
<super.tokenRef(...)>
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
<tokenRef(...)>
<listLabelElem(elem=label,...)>
<else>
<super.tokenRefAndListLabel(...)>
<endif>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.text)>,<terminalOptions.text; format="string"><endif>;
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.text)>,<terminalOptions.text; format="string"><endif>;
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
<else>
<super.tokenRefRuleRoot(...)>

```



```

<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
<tokenRefRuleRoot(...)>
<listLabelElem(elem=label,...)>
<else>
<super.tokenRefRuleRootAndListLabel(...)>
<endif>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.wildcard(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.DupTree(<label>);
adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>;
<endif>
<else>
<super.wildcard(...)>
<endif>
>>

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);

```

```

<endif><\n>
adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>
<else>
<super.matchSet(...)>
<endif>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<if(!ruleDescriptor.isSynPred)>
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
<else>
<super.matchRuleBlockSet(...)>
<endif>
>>

matchSetBang(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(...)>
<else>
<super.matchSetBang(...)>
<endif>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<if(!ruleDescriptor.isSynPred)>
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>\}<endif>
<endif>
}, ...
)>

```

```

<else>
<super.matchSetRuleRoot(...)>
<endif>
>>

// RULE REF AST

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
<if(!rewriteMode)>
adaptor.AddChild(root_<treeLevel>, <label>.Tree);
<else> <! rewrite mode !>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>.Tree;
<endif>
<else>
<super.ruleRef(...)>
<endif>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<ruleRef(...)>
<listLabelElem(elem={<label>.Tree},...)>
<else>
<super.ruleRefAndListLabel(...)>
<endif>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>root_<treeLevel> =
(<ASTLabelType>)adaptor.BecomeRoot(<label>.Tree, root_<treeLevel>);
<endif>
<else>
<super.ruleRefRuleRoot(...)>
<endif>
>>

/** ^(x+=rule ...) auto construct */

```

```

ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<ruleRefRuleRoot(...)>
<listLabelElem(elem={<label>.Tree},...)>
<else>
<super.ruleRefRuleRootAndListLabel(...)>
<endif>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrackAndListLabel(...)>
<else>
<super.ruleRefTrackAndListLabel(...)>
<endif>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRootTrack(...)>
<else>
<super.ruleRefRuleRootTrack(...)>
<endif>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRuleRootTrackAndListLabel(...)>
<else>
<super.ruleRefRuleRootTrackAndListLabel(...)>
<endif>
>>

/** Streams for token refs are tree nodes now; override to
* change NextToken to NextNode.
*/

```

```

createRewriteNodeFromElement(token,terminalOptions,args) ::= <%
<if(terminalOptions.node)>
new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif>stream_<token>.NextNode())
<else>
stream_<token>.NextNode()
<endif>
%>

```

```

ruleCleanup() ::= <<
<super.ruleCleanup()>
<if(!ruleDescriptor.isSynPred)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
retval.Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
<if(backtracking)>}<endif>
<endif>
<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/ASTTreeParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2006 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
(INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

finishedBacktracking(block) ::= <<

<if(backtracking)>

if <actions.(actionScope).synpredgate>:

<block>

<else>

<block>

<endif>

>>

/** Add a variable to track last element matched */

ruleDeclarations() ::= <<

<super.ruleDeclarations()>

_first_0 = None

_last = None<\n>

>>

/** What to emit when there is no rewrite rule. For auto build

* mode, does nothing.

*/

noRewrite(rewriteBlockLevel, treeLevel) ::= <<

<finishedBacktracking({

<if(rewriteMode)>

retval.tree = _first_0

if self._adaptor.getParent(retval.tree) is not None and self._adaptor.isNil(self._adaptor.getParent(retval.tree)):

retval.tree = self._adaptor.getParent(retval.tree)

<endif>

})>

>>

/** match ^(root children) in tree parser; override here to

* add tree construction actions.

```

*/
tree(root, actionsAfterRoot, children, nullableChildList,
  enclosingTreeLevel, treeLevel) ::= <<
  _last = self.input.LT(1)
  _save_last_<treeLevel> = _last
  _first_<treeLevel> = None
  <if(!rewriteMode)>
  root_<treeLevel> = self._adaptor.nil()<\n>
  <endif>
  <root:element()>
  <if(rewriteMode)>
  <finishedBacktracking({
  <if(root.el.rule)>
  if _first_<enclosingTreeLevel> is None:
    _first_<enclosingTreeLevel> = <root.el.label>.tree<\n>
  <else>
  if _first_<enclosingTreeLevel> is None:
    _first_<enclosingTreeLevel> = <root.el.label><\n>
  <endif>
  })>
  <endif>
  <actionsAfterRoot:element()>
  <if(nullableChildList)>
  if self.input.LA(1) == DOWN:
    self.match(self.input, DOWN, None)
    <children:element()>
    self.match(self.input, UP, None)

  <else>
  self.match(self.input, DOWN, None)
  <children:element()>
  self.match(self.input, UP, None)<\n>
  <endif>
  <if(!rewriteMode)>
  self._adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>)<\n>
  <endif>
  _last = _save_last_<treeLevel>

  >>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions={}) ::= <<
  _last = self.input.LT(1)
  <super.tokenRef(...)>

```

>>

```
/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
  _last = self.input.LT(1)
  <super.tokenRef(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  <if(terminalOptions.node)>
  <label>_tree = <terminalOptions.node>(<label>)
  <else>
  <label>_tree = self._adaptor.dupNode(<label>)
  <endif><\n>
  self._adaptor.addChild(root_<treeLevel>, <label>_tree)
  })>
  <else> <! rewrite mode !>
  <finishedBacktracking({
  if _first_<treeLevel> is None:
    _first_<treeLevel> = <label><\n>
  })>
  <endif>
>>
```

```
/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
  <tokenRef(...)>
  <listLabel(elem=label,...)>
>>
```

```
/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={}) ::= <<
  _last = self.input.LT(1)
  <super.tokenRef(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  <if(terminalOptions.node)>
  <label>_tree = <terminalOptions.node>(<label>)
  <else>
  <label>_tree = self._adaptor.dupNode(<label>)
  <endif><\n>
  root_<treeLevel> = self._adaptor.becomeRoot(<label>_tree, root_<treeLevel>)
  })>
  <endif>
>>
```

```
/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
  <tokenRefRuleRoot(...)>
```



```

<listLabel(elem=label,...)>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.wildcard(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  <label>_tree = self._adaptor.dupTree(<label>)
  self._adaptor.addChild(root_<treeLevel>, <label>_tree)
  })>
  <else> <! rewrite mode !>
  <finishedBacktracking({
  if _first_<treeLevel> is None:
    _first_<treeLevel> = <label>
  })>
  <endif>
>>

// SET AST
matchSet(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.matchSet(postmatchCode={
  <if(!rewriteMode)>
  <finishedBacktracking({
  <if(terminalOptions.node)>
  <label>_tree = <terminalOptions.node>(<label>)
  <else>
  <label>_tree = self._adaptor.dupNode(<label>)
  <endif><\n>
  self._adaptor.addChild(root_<treeLevel>, <label>_tree)
  })>
  <endif>
  }, ...)>
>>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={ }) ::= <<
  <matchSet(...)>
  <noRewrite(...)> <! set return tree !>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.matchSet(...)>
>>

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={ }) ::= <<

```

```

<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<finishedBacktracking({
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>(<label>)
<else>
<label>_tree = self._adaptor.dupNode(<label>)
<endif><\n>
root_<treeLevel> = self._adaptor.becomeRoot(<label>_tree, root_<treeLevel>)
})>
<endif>
}, ...)>
>>

```

// RULE REF AST

```

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
  _last = self.input.LT(1)
  <super.ruleRef(...)>
  <finishedBacktracking({
  <if(!rewriteMode)>
  self._adaptor.addChild(root_<treeLevel>, <label>.tree)
  <else> <! rewrite mode !>
  if _first_<treeLevel> is None:
    _first_<treeLevel> = <label>.tree<\n>
  <endif>
  })>
>>

```

```

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
  <ruleRef(...)>
  <listLabel(label, {<label>.tree})>
>>

```

```

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
  _last = self.input.LT(1)
  <super.ruleRef(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  root_<treeLevel> = self._adaptor.becomeRoot(<label>.tree, root_<treeLevel>)
  })>
  <endif>
>>

```

```

/** ^(x+=rule ...) auto construct */

```

```

ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(label, {<label>.tree})>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefTrackAndListLabel(...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefRuleRootTrack(...)>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefRuleRootTrackAndListLabel(...)>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,scope,terminalOptions={}) ::= <<
<if(terminalOptions.node)>
<terminalOptions.node>(stream_<token>.nextNode())
<else>
stream_<token>.nextNode()
<endif>
>>

ruleCleanup() ::= <<
<super.ruleCleanup()>
<if(!rewriteMode)>
<finishedBacktracking({
retval.tree = self._adaptor.rulePostProcessing(root_0)
})>
<endif>
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Python/ASTTreeParser.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2010 Terence Parr

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/MultiMap.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/Utils.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/RuleLabelScope.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/Python3Target.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/StateCluster.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/IntervalSet.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarSerializerFoo.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/DFAOptimizer.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-

jar/org/antlr/tool/GrammarSyntaxMessage.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/NFAFactory.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Message.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFAState.java

- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/CTarget.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarUnreachableAltsMessage.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/Tool.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFAContext.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Grammar.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFAToDFAConverter.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/NameSpaceChecker.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarReport2.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/IntArrayList.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/DelphiTarget.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/AnalysisTimeoutException.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/Perl5Target.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/LookaheadSet.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/BitSet.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/LL1Analyzer.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFA.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/MutableInteger.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarSpelunker.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/ActionScriptTarget.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarDanglingStateMessage.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/BuildDependencyGenerator.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/CompositeGrammar.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/DecisionProbe.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/CSharp2Target.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/CompositeGrammarTree.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFAConfiguration.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Interpreter.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/SemanticContext.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/JavaScriptTarget.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarSemanticsMessage.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarInsufficientPredicatesMessage.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NFAConversionThread.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/RuleClosureTransition.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Rule.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Attribute.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/Barrier.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/PythonTarget.java
- * /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/DOTGenerator.java

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/ToolMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/RandomPhrase.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Strip.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/State.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/AnalysisRecursionOverflowException.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/Label.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/LeftRecursionCyclesMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/CppTarget.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/AssignTokenTypesBehavior.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/AttributeScope.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/Interval.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarNonDeterminismMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/DFASState.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/IntSet.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/Transition.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/Target.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/Interp.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/LL1DFA.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/RecursionOverflowMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarAnalysisAbortedMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/ANTLRErrorListener.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/NonRegularDecisionMessage.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarSanity.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/PredicateLabel.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/NonLLStarDecisionException.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/ACyclicDFACodeGenerator.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/JavaTarget.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarAST.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/GrammarReport.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/ActionLabel.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/Graph.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/ErrorMessageManager.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/DFA.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/FASerializer.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/misc/OrderedHashSet.java
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/analysis/MachineProbe.java
No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2005-2012 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

```
finishedBacktracking(block) ::= <<
```

```
<if(backtracking)>
```

```
if <actions.(actionScope).synpredgate>:
```

```
  <block>
```

```
<else>
```

```
<block>
```

```
<endif>
```

```
>>
```

```

/** Add a variable to track last element matched */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
_first_0 = None
_last = None<\n>
>>

/** What to emit when there is no rewrite rule. For auto build
* mode, does nothing.
*/
noRewrite(rewriteBlockLevel, treeLevel) ::= <<
<finishedBacktracking({
<if(rewriteMode)>
retval.tree = _first_0
if self._adaptor.getParent(retval.tree) is not None and self._adaptor.isNil(self._adaptor.getParent(retval.tree)):
    retval.tree = self._adaptor.getParent(retval.tree)
<endif>
})>
>>

/** match ^(root children) in tree parser; override here to
* add tree construction actions.
*/
tree(root, actionsAfterRoot, children, nullableChildList,
    enclosingTreeLevel, treeLevel) ::= <<
_last = self.input.LT(1)
_save_last_<treeLevel> = _last
_first_<treeLevel> = None
<if(!rewriteMode)>
root_<treeLevel> = self._adaptor.nil()<\n>
<endif>
<root:element()>
<if(rewriteMode)>
<finishedBacktracking({
<if(root.el.rule)>
if _first_<enclosingTreeLevel> is None:
    _first_<enclosingTreeLevel> = <root.el.label>.tree<\n>
<else>
if _first_<enclosingTreeLevel> is None:
    _first_<enclosingTreeLevel> = <root.el.label><\n>
<endif>
})>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if self.input.LA(1) == DOWN:
    self.match(self.input, DOWN, None)
<children:element()>

```



```

self.match(self.input, UP, None)

<else>
self.match(self.input, DOWN, None)
<children:element()>
self.match(self.input, UP, None)<\n>
<endif>
<if(!rewriteMode)>
self._adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>)<\n>
<endif>
_last = _save_last_<treeLevel>

>>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions={ }) ::= <<
_last = self.input.LT(1)
<super.tokenRef(...)>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions={ }) ::= <<
_last = self.input.LT(1)
<super.tokenRef(...)>
<if(!rewriteMode)>
<finishedBacktracking({
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>(<label>)
<else>
<label>_tree = self._adaptor.dupNode(<label>)
<endif><\n>
self._adaptor.addChild(root_<treeLevel>, <label>_tree)
})>
<else> <! rewrite mode !>
<finishedBacktracking({
if _first_<treeLevel> is None:
_first_<treeLevel> = <label><\n>
})>
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<tokenRef(...)>

```

```

<listLabel(elem=label,...)>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.tokenRef(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  <if(terminalOptions.node)>
  <label>_tree = <terminalOptions.node>(<label>)
  <else>
  <label>_tree = self._adaptor.dupNode(<label>)
  <endif><\n>
  root_<treeLevel> = self._adaptor.becomeRoot(<label>_tree, root_<treeLevel>)
  })>
  <endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
  <tokenRefRuleRoot(...)>
  <listLabel(elem=label,...)>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.wildcard(...)>
  <if(!rewriteMode)>
  <finishedBacktracking({
  <label>_tree = self._adaptor.dupTree(<label>)
  self._adaptor.addChild(root_<treeLevel>, <label>_tree)
  })>
  <else> <! rewrite mode !>
  <finishedBacktracking({
  if _first_<treeLevel> is None:
    _first_<treeLevel> = <label>
  })>
  <endif>
>>

// SET AST
matchSet(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= <<
  _last = self.input.LT(1)
  <super.matchSet(postmatchCode={
  <if(!rewriteMode)>
  <finishedBacktracking({

```

```

<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>( <label>)
<else>
<label>_tree = self._adaptor.dupNode(<label>)
<endif><\n>
self._adaptor.addChild(root_<treeLevel>, <label>_tree)
})>
<endif>
}, ...>
>>

```

```

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={}) ::= <<
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
>>

```

```

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={}) ::= <<
_last = self.input.LT(1)
<super.matchSet(...)>
>>

```

```

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={}) ::= <<
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<finishedBacktracking({
<if(terminalOptions.node)>
<label>_tree = <terminalOptions.node>( <label>)
<else>
<label>_tree = self._adaptor.dupNode(<label>)
<endif><\n>
root_<treeLevel> = self._adaptor.becomeRoot(<label>_tree, root_<treeLevel>)
})>
<endif>
}, ...)>
>>

```

// RULE REF AST

```

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRef(...)>
<finishedBacktracking({
<if(!rewriteMode)>
self._adaptor.addChild(root_<treeLevel>, <label>.tree)
<else> <! rewrite mode !>
if _first_<treeLevel> is None:
_first_<treeLevel> = <label>.tree<\n>

```

```

<endif>
})>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(label, {<label>.tree})>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRef(...)>
<if(!rewriteMode)>
<finishedBacktracking({
root_<treeLevel> = self._adaptor.becomeRoot(<label>.tree, root_<treeLevel>)
})>
<endif>
>>

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(label, {<label>.tree})>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefTrack(...)>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefTrackAndListLabel(...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)
<super.ruleRefRootTrack(...)>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
_last = self.input.LT(1)

```

```

<super.ruleRefRuleRootTrackAndListLabel(...)>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,scope,terminalOptions={}) ::= <<
<if(terminalOptions.node)>
<terminalOptions.node>(stream_<token>.nextNode())
<else>
stream_<token>.nextNode()
<endif>
>>

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(!rewriteMode)>
<finishedBacktracking({
retval.tree = self._adaptor.rulePostProcessing(root_0)
})>
<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/Python3/ASTTreeParser.stg
```

No license file was found, but licenses were detected in source scan.

```

/*
[The "BSD license"]
Copyright (c) 2005-2006 Terence Parr
Copyright (c) 2008 Ronald Blaschke
All rights reserved.

```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

group ASTTreeParser;

/** Add a variable to track last element matched */

ruleDeclarations() ::= <<

<super.ruleDeclarations(>

<ASTLabelType> _first_0 = null;

<ASTLabelType> _last = null;<\n>

>>

/** What to emit when there is no rewrite rule. For auto build

* mode, does nothing.

*/

noRewrite(rewriteBlockLevel, treeLevel) ::= <<

<if(backtracking)>if (state.backtracking==0) {<endif>

<if(rewriteMode)>

retval.tree = (<ASTLabelType>)_first_0;

if (adaptor.getParent(retval.tree)!=null && adaptor.isNil(adaptor.getParent(retval.tree)))

retval.tree = (<ASTLabelType>)adaptor.getParent(retval.tree);

<endif>

<if(backtracking)>}<endif>

>>

/** match ^(root children) in tree parser; override here to

* add tree construction actions.

*/

tree(root, actionsAfterRoot, children, nullableChildList,

enclosingTreeLevel, treeLevel) ::= <<

_last = (<ASTLabelType>)input.LT(1);

```

{
<ASTLabelType> _save_last_<treeLevel> = _last;
<ASTLabelType> _first_<treeLevel> = null;
<if(!rewriteMode)>
<ASTLabelType> root_<treeLevel> = (<ASTLabelType>)adaptor.nil();
<endif>
<root:element()>
<if(rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 )<endif>
<if(root.el.rule)>
if ( _first_<enclosingTreeLevel>===null ) _first_<enclosingTreeLevel> = <root.el.label>.tree;
<else>
if ( _first_<enclosingTreeLevel>===null ) _first_<enclosingTreeLevel> = <root.el.label>;
<endif>
<endif>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if ( input.LA(1)==Token.DOWN ) {
  match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
  <children:element()>
  match(input, Token.UP, null); <checkRuleBacktrackFailure()>
}
<else>
match(input, Token.DOWN, null); <checkRuleBacktrackFailure()>
<children:element()>
match(input, Token.UP, null); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}<\n>
>>

```

```
// TOKEN AST STUFF
```

```

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
>>

```

```

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>

```

```

<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if ( state.backtracking==0 )<endif>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>;
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRoot(...)>
<listLabel(elem=label,...)>
>>

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(..., postmatchCode={

```



```

<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
adaptor.addChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<endif>
}
)>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<matchSet(...)>
<noRewrite()> <! set return tree !>
>>

matchSetBang(s,label,elementIndex,postmatchCode) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(...)>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<super.matchSet(..., postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) {<endif>
<if(terminalOptions.node)>
<label>_tree = new <terminalOptions.node>(<label>);
<else>
<label>_tree = (<ASTLabelType>)adaptor.dupNode(<label>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.becomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
}
)>
>>

// RULE REF AST

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(backtracking)>if ( state.backtracking==0 ) <endif>
<if(!rewriteMode)>

```

```

adaptor.addChild(root_<treeLevel>, <label>.getTree());
<else> <! rewrite mode !>
if ( _first_<treeLevel>==null ) _first_<treeLevel> = <label>.tree;
<endif>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label+".getTree()",...)>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) <endif>root_<treeLevel> =
(<ASTLabelType>)adaptor.becomeRoot(<label>.getTree(), root_<treeLevel>);
<endif>
>>

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabel(elem=label+".getTree()",...)>
>>

/** Streams for token refs are tree nodes now; override to
 * change nextToken to nextNode.
 */
createRewriteNodeFromElement(token,terminalOptions,scope) ::= <<
<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.nextNode())
<else>
stream_<token>.nextNode()
<endif>
>>

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(!rewriteMode)>
<if(backtracking)>if ( state.backtracking==0 ) {<\n><endif>
retval.tree = (<ASTLabelType>)adaptor.rulePostProcessing(root_0);
<if(backtracking)>}<endif>
<endif>
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Perl5/ASTTreeParser.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2006 Kay Roepke

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

New style messages. This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the format that mimicks GCC output.

*/

group gnu;

location(file, line, column) ::= "<file>:<line>:"

message(id, text) ::= "<text> (<id>)"

report(location, message, type) ::= "<location> <type>: <message>"

wantsSingleLineMessage() ::= "true"

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/tool/templates/messages/formats/gnu.stg

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2010 Terence Parr

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

New style messages. This file only contains the messages in English, but no information about which file, line, or column it occurred in.

The location and message ids are taken out of the formats directory.

Kay Roepke

*/

group en_US;

// TOOL ERRORS

// file errors

CANNOT_WRITE_FILE(arg,exception,stackTrace) ::= <<

cannot write file <arg>: <exception>

<stackTrace; separator="\n">

>>

CANNOT_CLOSE_FILE(arg,exception,stackTrace) ::= "cannot close file <arg>: <exception>"

CANNOT_FIND_TOKENS_FILE(arg) ::= "cannot find tokens file <arg>"

```

ERROR_READING_TOKENS_FILE(arg,exception,stackTrace) ::= <<
problem reading token vocabulary file <arg>: <exception>
<stackTrace; separator="\n">
>>
DIR_NOT_FOUND(arg,exception,stackTrace) ::= "directory not found: <arg>"
OUTPUT_DIR_IS_FILE(arg,exception,stackTrace) ::= "output directory is a file: <arg>"
CANNOT_OPEN_FILE(arg,exception,stackTrace) ::= "cannot find or open file: <arg><if(exception)>; reason:
<exception><endif>"
CIRCULAR_DEPENDENCY() ::= "your grammars contain a circular dependency and cannot be sorted into a valid
build order."

INTERNAL_ERROR(arg,arg2,exception,stackTrace) ::= <<
internal error: <arg> <arg2><if(exception)>: <exception><endif>
<stackTrace; separator="\n">
>>
INTERNAL_WARNING(arg) ::= "internal warning: <arg>"
ERROR_CREATING_ARTIFICIAL_RULE(arg,exception,stackTrace) ::= <<
problems creating lexer rule listing all tokens: <exception>
<stackTrace; separator="\n">
>>
TOKENS_FILE_SYNTAX_ERROR(arg,arg2) ::=
"problems parsing token vocabulary file <arg> on line <arg2>"
CANNOT_GEN_DOT_FILE(arg,exception,stackTrace) ::=
"cannot write DFA DOT file <arg>: <exception>"
BAD_ACTION_AST_STRUCTURE(exception,stackTrace) ::=
"bad internal tree structure for action '<arg>': <exception>"
BAD_AST_STRUCTURE(arg,exception,stackTrace) ::= <<
bad internal tree structure '<arg>': <exception>
<stackTrace; separator="\n">
>>
FILE_AND_GRAMMAR_NAME_DIFFER(arg,arg2) ::=
"file <arg2> contains grammar <arg>; names must be identical"
FILENAME_EXTENSION_ERROR(arg) ::=
"file <arg> must end in a file extension, normally .g"

// code gen errors
MISSING_CODE_GEN_TEMPLATES(arg) ::=
"cannot find code generation templates <arg>.stg"
MISSING_CYCLIC_DFA_CODE_GEN_TEMPLATES() ::=
"cannot find code generation cyclic DFA templates for language <arg>"
CODE_GEN_TEMPLATES_INCOMPLETE(arg) ::=
"at least one code generation template missing for language <arg>"
CANNOT_CREATE_TARGET_GENERATOR(arg,exception,stackTrace) ::=
"cannot create target <arg> code generator: <exception>"
STRING_TEMPLATE_ERROR(arg,exception,stackTrace) ::=
"template error: <arg>"
CANNOT_COMPUTE_SAMPLE_INPUT_SEQ() ::=
"cannot generate a sample input sequence from lookahead DFA"

```

```

// grammar interpretation errors
/*
NO_VIABLE_DFA_ALT(arg,arg2) ::=
    "no viable transition from state <arg> on <arg2> while interpreting DFA"
*/

// GRAMMAR ERRORS
SYNTAX_ERROR(arg) ::= "syntax error: <arg>"
RULE_REDEFINITION(arg) ::=
    "rule <arg> redefinition"
LEXER_RULES_NOT_ALLOWED(arg) ::=
    "lexer rule <arg> not allowed in parser"
PARSER_RULES_NOT_ALLOWED(arg) ::=
    "parser rule <arg> not allowed in lexer"
CANNOT_FIND_ATTRIBUTE_NAME_IN_DECL(arg) ::=
    "cannot find an attribute name in attribute declaration"
NO_TOKEN_DEFINITION(arg) ::=
    "no lexer rule corresponding to token: <arg>"
UNDEFINED_RULE_REF(arg) ::=
    "reference to undefined rule: <arg>"
LITERAL_NOT_ASSOCIATED_WITH_LEXER_RULE(arg) ::=
    "literal has no associated lexer rule: <arg>"
CANNOT_ALIAS_TOKENS_IN_LEXER(arg) ::=
    "literals are illegal in lexer tokens{ } section: <arg>"
ATTRIBUTE_REF_NOT_IN_RULE(arg,arg2) ::=
    "reference to attribute outside of a rule: <arg><if(arg2)>.<arg2><endif>"
UNKNOWN_ATTRIBUTE_IN_SCOPE(arg,arg2) ::=
    "unknown attribute for <arg>: <arg2>"
UNKNOWN_RULE_ATTRIBUTE(arg,arg2) ::=
    "unknown attribute for rule <arg>: <arg2>"
UNKNOWN_SIMPLE_ATTRIBUTE(arg,args2) ::=
    "attribute is not a token, parameter, or return value: <arg>"
ISOLATED_RULE_SCOPE(arg) ::=
    "missing attribute access on rule scope: <arg>"
INVALID_RULE_PARAMETER_REF(arg,arg2) ::=
    "cannot access rule <arg>'s parameter: <arg2>"
INVALID_RULE_SCOPE_ATTRIBUTE_REF(arg,arg2) ::=
    "cannot access rule <arg>'s dynamically-scoped attribute: <arg2>"
SYMBOL_CONFLICTS_WITH_GLOBAL_SCOPE(arg) ::=
    "symbol <arg> conflicts with global dynamic scope with same name"
WRITE_TO_READONLY_ATTR(arg,arg2,arg3) ::=
    "cannot write to read only attribute: $<arg><if(arg2)>.<arg2><endif>"
LABEL_CONFLICTS_WITH_RULE(arg) ::=
    "label <arg> conflicts with rule with same name"
LABEL_CONFLICTS_WITH_TOKEN(arg) ::=
    "label <arg> conflicts with token with same name"
LABEL_CONFLICTS_WITH_RULE_SCOPE_ATTRIBUTE(arg,arg2) ::=

```

"label <arg> conflicts with rule <arg2>'s dynamically-scoped attribute with same name"

LABEL_CONFLICTS_WITH_RULE_ARG_RETVAL(arg,arg2) ::=

"label <arg> conflicts with rule <arg2>'s return value or parameter with same name"

ATTRIBUTE_CONFLICTS_WITH_RULE(arg,arg2) ::=

"rule <arg2>'s dynamically-scoped attribute <arg> conflicts with the rule name"

ATTRIBUTE_CONFLICTS_WITH_RULE_ARG_RETVAL(arg,arg2) ::=

"rule <arg2>'s dynamically-scoped attribute <arg> conflicts with<arg2>'s return value or parameter with same name"

LABEL_TYPE_CONFLICT(arg,arg2) ::=

"label <arg> type mismatch with previous definition: <arg2>"

ARG_RETVAL_CONFLICT(arg,arg2) ::=

"rule <arg2>'s argument <arg> conflicts a return value with same name"

NONUNIQUE_REF(arg) ::=

"<arg> is a non-unique reference"

FORWARD_ELEMENT_REF(arg) ::=

"illegal forward reference: <arg>"

MISSING_RULE_ARGS(arg) ::=

"missing parameter(s) on rule reference: <arg>"

RULE_HAS_NO_ARGS(arg) ::=

"rule <arg> has no defined parameters"

ARGS_ON_TOKEN_REF(arg) ::=

"token reference <arg> may not have parameters"

ILLEGAL_OPTION(arg) ::=

"illegal option <arg>"

LIST_LABEL_INVALID_UNLESS_RETVAL_STRUCT(arg) ::=

"rule '+=' list labels are not allowed w/o output option: <arg>"

UNDEFINED_TOKEN_REF_IN_REWRITE(arg) ::=

"reference to undefined token in rewrite rule: <arg>"

REWRITE_ELEMENT_NOT_PRESENT_ON_LHS(arg) ::=

"reference to rewrite element <arg> without reference on left of ->"

UNDEFINED_LABEL_REF_IN_REWRITE(arg) ::=

"reference to undefined label in rewrite rule: \$<arg>"

NO_GRAMMAR_START_RULE (arg) ::=

"grammar <arg>: no start rule (no rule can obviously be followed by EOF)"

EMPTY_COMPLEMENT(arg) ::= <<

<if(arg)>

set complement ~<arg> is empty

<else>

set complement is empty

<endif>

>>

UNKNOWN_DYNAMIC_SCOPE(arg) ::=

"unknown dynamic scope: <arg>"

UNKNOWN_DYNAMIC_SCOPE_ATTRIBUTE(arg,arg2) ::=

"unknown dynamically-scoped attribute for scope <arg>: <arg2>"

RULE_REF_AMBIG_WITH_RULE_IN_ALT(arg) ::=

"reference \$<arg> is ambiguous; rule <arg> is enclosing rule and referenced in the production (assuming enclosing rule)"

ISOLATED_RULE_ATTRIBUTE(arg) ::=
 "reference to locally-defined rule scope attribute without rule name: <arg>"

INVALID_ACTION_SCOPE(arg,arg2) ::=
 "unknown or invalid action scope for <arg2> grammar: <arg>"

ACTION_REDEFINITION(arg) ::=
 "redefinition of <arg> action"

DOUBLE_QUOTES_ILLEGAL(arg) ::=
 "string literals must use single quotes (such as '\begin\'): <arg>"

INVALID_TEMPLATE_ACTION(arg) ::=
 "invalid StringTemplate % shorthand syntax: '<arg>'"

MISSING_ATTRIBUTE_NAME() ::=
 "missing attribute name on \$ reference"

ARG_INIT_VALUES_ILLEGAL(arg) ::=
 "rule parameters may not have init values: <arg>"

REWRITE_OR_OP_WITH_NO_OUTPUT_OPTION(arg) ::=
 "<if(arg)>rule <arg> uses <endif>rewrite syntax or operator with no output option; setting output=AST"

AST_OP_WITH_NON_AST_OUTPUT_OPTION(arg) ::=
 "AST operator with non-AST output option: <arg>"

NO_RULES(arg) ::= "grammar file <arg> has no rules"

MISSING_AST_TYPE_IN_TREE_GRAMMAR(arg) ::=
 "tree grammar <arg> has no ASTLabelType option"

REWRITE_FOR_MULTI_ELEMENT_ALT(arg) ::=
 "with rewrite=true, alt <arg> not simple node or obvious tree element; text attribute for rule not guaranteed to be correct"

RULE_INVALID_SET(arg) ::=
 "Cannot complement rule <arg>; not a simple set or element"

HETERO_ILLEGAL_IN_REWRITE_ALT(arg) ::=
 "alts with rewrites can't use heterogeneous types left of ->"

NO_SUCH_GRAMMAR_SCOPE(arg,arg2) ::=
 "reference to undefined grammar in rule reference: <arg>.<arg2>"

NO_SUCH_RULE_IN_SCOPE(arg,arg2) ::=
 "rule <arg2> is not defined in grammar <arg>"

TOKEN_ALIAS_CONFLICT(arg,arg2) ::=
 "cannot alias <arg>; string already assigned to <arg2>"

TOKEN_ALIAS_REASSIGNMENT(arg,arg2) ::=
 "cannot alias <arg>; token name already assigned to <arg2>"

TOKEN_VOCAB_IN_DELEGATE(arg,arg2) ::=
 "tokenVocab option ignored in imported grammar <arg>"

INVALID_IMPORT(arg,arg2) ::=
 "<arg.grammarTypeString> grammar <arg.name> cannot import <arg2.grammarTypeString> grammar <arg2.name>"

IMPORTED_TOKENS_RULE_EMPTY(arg,arg2) ::=
 "no lexer rules contributed to <arg> from imported grammar <arg2>"

IMPORT_NAME_CLASH(arg,arg2) ::=
 "combined grammar <arg.name> and imported <arg2.grammarTypeString> grammar <arg2.name> both generate <arg2.recognizerName>; import ignored"

AST_OP_IN_ALT_WITH_REWRITE(arg,arg2) ::=
 "rule <arg> alt <arg2> uses rewrite syntax and also an AST operator"

WILDCARD_AS_ROOT(arg) ::= "Wildcard invalid as root; wildcard can itself be a tree"
CONFLICTING_OPTION_IN_TREE_FILTER(arg,arg2) ::= "option <arg>=<arg2> conflicts with tree grammar filter mode"
ILLEGAL_OPTION_VALUE(arg, arg2) ::= "value '<arg2>' invalid for option <arg>"
ALL_OPS_NEED_SAME_ASSOC(arg) ::= "all operators of alt <alt> of left-recursive rule must have same associativity"
RANGE_OP_ILLEGAL(arg) ::= "the .. range operator isn't allowed in parser rules"

// GRAMMAR WARNINGS

GRAMMAR_NONDETERMINISM(input,conflictingAlts,paths,disabled,hasPredicateBlockedByAction) ::= <<
<if(paths)>
Decision can match input such as "<input>" using multiple alternatives:
<paths:{ it | alt <it.alt> via NFA path <it.states; separator=","><\n>}>
<else>
Decision can match input such as "<input>" using multiple alternatives: <conflictingAlts; separator=",">
<endif>
<if(disabled)><\n>As a result, alternative(s) <disabled; separator=","> were disabled for that input<endif><if(hasPredicateBlockedByAction)><\n>Semantic predicates were present but were hidden by actions.<endif>
>>

DANGLING_STATE(danglingAlts,input) ::= <<
the decision cannot distinguish between alternative(s) <danglingAlts; separator=","> for input such as "<input>"
>>

UNREACHABLE_ALTS(alts) ::= <<
The following alternatives can never be matched: <alts; separator=","><\n>
>>

INSUFFICIENT_PREDICATES(upon,altToLocations,hasPredicateBlockedByAction) ::= <<
Input such as "<upon>" is insufficiently covered with predicates at locations: <altToLocations.keys:{ alt|alt <alt>:
<altToLocations.(alt):{loc| line <loc.line>:<loc.column> at <loc.text>}>; separator=",">; separator=",">
>><if(hasPredicateBlockedByAction)><\n>Semantic predicates were present but were hidden by actions.<endif>
>>

DUPLICATE_SET_ENTRY(arg) ::=
"duplicate token type <arg> when collapsing subrule into set"

ANALYSIS_ABORTED(enclosingRule) ::= <<
ANTLR could not analyze this decision in rule <enclosingRule>; often this is because of recursive rule references visible from the left edge of alternatives. ANTLR will re-analyze the decision with a fixed lookahead of k=1. Consider using "options {k=1;}" for that decision and possibly adding a syntactic predicate.
>>

RECURSION_OVERFLOW(alt,input,targetRules,callSiteStates) ::= <<
Alternative <alt>: after matching input such as <input> decision cannot predict what comes next due to recursion

```
overflow <targetRules,callSiteStates:{t,c|to <|> from <c:{s|<s.enclosingRule.name>};separator=", ">}; separator="
and ">
>>
```

```
LEFT_RECURSION(targetRules,alt,callSiteStates) ::= <<
Alternative <alt> discovers infinite left-recursion <targetRules,callSiteStates:{t,c|to <|> from
<c:{s|<s.enclosingRule>};separator=", ">}; separator=" and ">
>>
```

```
UNREACHABLE_TOKENS(tokens) ::= <<
The following token definitions can never be matched because prior tokens match the same input: <tokens;
separator=", ">
>>
```

```
TOKEN_NONDETERMINISM(input,conflictingTokens,paths,disabled,hasPredicateBlockedByAction) ::=
<<
<if(paths)>
Decision can match input such as "<input>" using multiple alternatives:
<paths:{ it | alt <it.alt> via NFA path <it.states; separator=", "><\n>}>
<else>
Multiple token rules can match input such as "<input>": <conflictingTokens; separator=", "><\n>
<endif>
<if(disabled)><\n>As a result, token(s) <disabled; separator=", "> were disabled for that
input<endif><if(hasPredicateBlockedByAction)><\n>Semantic predicates were present but were hidden by
actions.<endif>
>>
```

```
LEFT_RECURSION_CYCLES(listOfCycles) ::= <<
The following sets of rules are mutually left-recursive <listOfCycles:{c| [<c:{r|<r.name>}; separator=", ">];
separator=" and ">
>>
```

```
NONREGULAR_DECISION(ruleName,alts) ::= <<
[fatal] rule <ruleName> has non-LL(*) decision due to recursive rule invocations reachable from alts <alts;
separator=", ">. Resolve by left-factoring or using syntactic predicates or using backtrack=true option.
>>
```

```
/* 110n for message levels */
warning() ::= "warning"
error() ::= "error"
```

```
Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/tool/templates/messages/languages/en.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
* [The "BSD license"]
```

```

* Copyright (c) 2010 Terence Parr
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
*   notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
*   notice, this list of conditions and the following disclaimer in the
*   documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
*   derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
/** ANTLR's code generator.
*
* Generate recognizers derived from grammars. Language independence
* achieved through the use of STGroup objects. All output
* strings are completely encapsulated in the group files such as Java.stg.
* Some computations are done that are unused by a particular language.
* This generator just computes and sets the values into the templates;
* the templates are free to use or not use the information.
*
* To make a new code generation target, define X.stg for language X
* by copying from existing Y.stg most closely related to your language;
* e.g., to do CSharp.stg copy Java.stg. The template group file has a
* bunch of templates that are needed by the code generator. You can add
* a new target w/o even recompiling ANTLR itself. The language=X option
* in a grammar file dictates which templates get loaded/used.
*
* Some language like C need both parser files and header files. Java needs
* to have a separate file for the cyclic DFA as ANTLR generates bytecodes
* directly (which cannot be in the generated parser Java file). To facilitate
* this,
*
* cyclic can be in same file, but header, output must be searpate. recognizer
* is in outptufile.

```

*/

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/CodeGenerator.java

No license file was found, but licenses were detected in source scan.

/*

[The "BSD license"]

Copyright (c) 2006 Kay Roepke

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

New style messages. This file contains the actual layout of the messages emitted by ANTLR.

The text itself is coming out of the languages/*stg files, according to the chosen locale.

This file contains the default format ANTLR uses.

*/

group antlr;

location(file, line, column) ::= "<file>(<line>,<column>)"

message(id, text) ::= "error <id> : <text>"

report(location, message, type) ::= "<location> : <type> <message.id> : <message.text>"

wantsSingleLineMessage() ::= "true"

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/tool/templates/messages/formats/vs2005.stg
```

No license file was found, but licenses were detected in source scan.

group Dbg;

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2009 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
@parserBody.mixins() ::= <<
```

```
include ANTLR3::<if(profile)>Profile<else>Debug<endif>::ParserEvents
```

```
>>
```

```
@parserBody.additionalMembers() ::= <<
```

```
<if(grammar.grammarIsRoot)>
```

```
RULE_NAMES = [
```

```
:invalid_rule, <grammar.allImportedRules:{rST|:<rST.name>}; wrap="\n ", separator=", ">
```

```
].freeze
```

```
<endif>
```

```
>>
```

```

@rule.body() ::= <<
in_rule(grammar_file_name, "<ruleName>") do
  @debug_listener.location(<ruleDescriptor.tree.line>, <ruleDescriptor.tree.column>)

  <@super.body()>

  @debug_listener.location(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.column>)
end
>>

// Common debug event triggers used by region overrides below
enterSubRule() ::= <<
in_subrule(<decisionNumber>) do<\n>
>>

exitSubRule() ::= <<
end<\n>
>>

enterDecision() ::= <<
in_decision(<decisionNumber>) do<\n>
>>

exitDecision() ::= <<
end<\n>
>>

enterAlt(n) ::= <<
in_alternative(<n>)<\n>
>>

// Region overrides that tell various constructs to add debugging triggers
@block.body() ::= <<
in_subrule(<decisionNumber>) do
  <@super.body()>
end
>>

// @blockBody.predecision() ::= "<enterSubRule()>"

```

```
// @blockBody.postdecision() ::= "<exitDecision>"
// @blockBody.postbranch() ::= "<exitSubRule>"
@blockBody.decision() ::= <<
in_decision(<decisionNumber>) do
  <@super.decision()>
end
>>
```

```
@ruleBlock.decision() ::= <<
in_decision(<decisionNumber>) do
  <@super.decision()>
end<\n>
>>
```

```
@ruleBlockSingleAlt.prealt() ::= "<enterAlt(n={ 1 })>"
```

```
@blockSingleAlt.prealt() ::= "<enterAlt(n={ 1 })>"
```

```
@positiveClosureBlock.loopBody() ::= <<
in_subrule(<decisionNumber>) do
  <@super.loopBody()>
end
>>
```

```
@positiveClosureBlockLoop.decisionBody() ::= <<
in_decision(<decisionNumber>) do
  <@super.decisionBody()>
end
>>
```

```
@positiveClosureBlockLoop.earlyExitException() ::= <<
@debug_listener.recognition_exception(eee)
>>
```

```
@closureBlock.loopBody() ::= <<
in_subrule(<decisionNumber>) do
  <@super.loopBody()>
end
>>
```

```

@closureBlockLoop.decisionBody() ::= <<
in_decision(<decisionNumber>) do
  <@super.decisionBody()>
end
>>

```

```

@altSwitchCase.preatt() ::= "<enterAlt(altNum)>" // altNum is arg of altSwitchCase

```

```

element(e) ::= <<
@debug_listener.location(<e.line>, <e.pos>) // e is arg of element
<super.element(e)>
>>

```

```

@matchSet.mismatchedSetException() ::= "@debug_listener.recognition_exception(mse)"

```

```

@dfaState.noViableAltException() ::= <<
nvae = NoViableAlternative( "<description>", <decisionNumber>, <stateNumber> )
@debug_listener.recognition_exception( nvae )
raise( nvae )<\n>
>>

```

```

@dfaStateSwitch.noViableAltException() ::= <<
nvae = NoViableAlternative( "<description>", <decisionNumber>, <stateNumber> )
@debug_listener.recognition_exception( nvae )
raise( nvae )<\n>
>>

```

```

dfaDecision(decisionNumber,description) ::= <<
begin
  @state.cyclic_decision = true
  <super.dfaDecision(...)>
rescue ANTLR3::Error::NoViableAlternative => nvae
  @debug_listener.recognition_exception(nvae)
  raise
end
>>

```

```

@cyclicDFA.errorMethod() ::= <<
def error(nvae)
  @recognizer.debug_listener.recognition_exception(nvae)
  super
end
>>

```



```
/** Force predicate validation to trigger an event */
evalPredicate(pred,description) ::= <<
predicate?("<description>") { <pred> }
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Ruby/Dbg.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2010 Kyle Yetter
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/RubyTarget.java
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
```

```
http://www.temporal-wave.com
```

```
http://www.linkedin.com/in/jimidle
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/
```

```
/** Add an adaptor property that knows how to build trees */
```

```
@headerFile.members() ::= <<
```

```
/* @headerFile.members() */
```

```
pANTLR3_BASE_TREE_ADAPTOR adaptor;
```

```
pANTLR3_VECTOR_FACTORY vectors;
```

```
/* End @headerFile.members() */
```

```
>>
```

```
/** Install the tree adaptor interface pointer and anything else that
```

```
* tree parsers and producers require.
```

```
*/
```

```
@genericParser.apifuncs() ::= <<
```

```
<if(PARSER)>
```

```
ADAPTOR = ANTLR3_TREE_ADAPTORNew(instream->tstream->tokenSource->strFactory);<\n>
```

```
<endif>
```

```
ctx->vectors = antlr3VectorFactoryNew(0);
```

```
>>
```

```
@genericParser.cleanup() ::= <<
```

```
ctx->vectors->close(ctx->vectors);
```

```
<if(PARSER)>
```

```
/* We created the adaptor so we must free it
```

```
*/
```

```
ADAPTOR->free(ADAPTOR);
```

```
<endif>
```

```

>>

@returnScope.ruleReturnMembers() ::= <<

<super.ASTLabelType()> tree;

>>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
<ASTLabelType> root_0;<\n>
>>

ruleInitializations() ::= <<
<super.ruleInitializations()>
root_0 = NULL;<\n>
>>

ruleLabelDefs() ::= <<
<super.ruleLabelDefs()>
<ruleDescriptor.tokenLabels:{it | <ASTLabelType> <it.label.text>_tree;}; separator="\n">
<ruleDescriptor.tokenListLabels:{it | <ASTLabelType> <it.label.text>_tree;}; separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites
  :{it | pANTLR3_REWRITE_RULE_<rewriteElementType>_STREAM stream_<it>;}; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
  :{it | pANTLR3_REWRITE_RULE_SUBTREE_STREAM stream_<it>;}; separator="\n">
>>

/* Note that we defer the actual creation of any rewrite streams we need here and just initialize
 * them to NULL. This saves creating huge numbers of rewrite streams that cannot be used as only
 * one alt will be taken in a rule, but we are declaring all the streams here. So we define
 * a macro that contains the create code, then use this macro later to check if the stream
 * has been created yet. Checking for NULL is almost free in C.
 */
ruleLabelInitializations() ::= <<
<super.ruleLabelInitializations()>
<ruleDescriptor.tokenLabels:{it | <it.label.text>_tree = NULL;}; separator="\n">
<ruleDescriptor.tokenListLabels:{it | <it.label.text>_tree = NULL;}; separator="\n">

<ruleDescriptor.allTokenRefsInAltsWithRewrites
  :{it | stream_<it> = NULL;
  #define CREATE_stream_<it> if (stream_<it> == NULL) {stream_<it> =
  antlr3RewriteRule<rewriteElementType>StreamNewAE(ADAPTOR, RECOGNIZER, (pANTLR3_UINT8)"token
  <it>"); \} }; separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
  :{it | stream_<it> = NULL;
  #define CREATE_stream_<it> if (stream_<it> == NULL) {stream_<it> =

```

```
antlr3RewriteRuleSubtreeStreamNewAE(ADAPTOR, RECOGNIZER, (pANTLR3_UINT8)"rule <it>"); \}};
separator="\n">
```

```
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.tree = NULL;
<endif>
>>
```

```
/** a rule label including default value */
ruleLabelInitVal(label) ::= <<
<super.ruleLabelInitVal(...)>
<label.label.text>.tree = NULL;
>>
```

```
/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<endif>
<endif>
>>
```

```
@alt.initializations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
root_0 = (<ASTLabelType>)(ADAPTOR->nilNode(ADAPTOR));<\n>
<endif>
<endif>
<endif>
>>
```

```
// Tracking Rule Elements
//
```

```
/** ID but track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) { <endif>CREATE_stream_<token>;
stream_<token>->add(stream_<token>, <label>, NULL);<if(backtracking)> }<endif><\n>
>>
```

```
/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
```

```

* to the tracking list stream_ID for use in the rewrite.
*/
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) { <endif>CREATE_stream_<token>; stream_<token>-
>add(stream_<token>, <label>, NULL);<if(backtracking)> }<endif><\n>
>>

wildcardTrack(label,elementIndex) ::= <<
<super.wildcard(...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) { <endif>CREATE_stream_<rule.name>;
stream_<rule.name>->add(stream_<rule.name>, <label>.tree, NULL);<if(backtracking)> }<endif>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabelTrack(...)>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) { <endif>CREATE_stream_<rule.name>;
stream_<rule.name>->add(stream_<rule.name>, <label>.tree, NULL);<if(backtracking)> }<endif>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabelAST(...)>
>>

// RULE REF AST

```

```

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label,...)>
>>

/* How to accumulate lists when we are doing rewrite tracking...
*/
listLabelTrack(label) ::= <<
/* listLabelTrack(label)
*/
if (list_<label> == NULL)
{
    list_<label>=ctx->vectors->newVector(ctx->vectors);
}
list_<label>->add(list_<label>, <label>.tree, NULL);
>>

/* How to accumulate lists of rule outputs (only allowed with AST
* option but if the user is going to walk the tree, they will want
* all their custom elements from rule returns.
*
* Normally, we use inline structures (which the compiler lays down
* code to copy from heap allocations. However, here we want to accumulate copies
* of the returned structures because we are adding them to a list. This only makes sense if the
* grammar is not rewriting the tree as a tree rewrite only preserves the tree, not the object/structure
* returned from the rule. The rewrite will extract the tree pointer. However, if we are not going to
* do a tree re-write, then the user may wish to iterate the structures returned by the rule in
* action code and will expect the user defined returns[] elements to be available when they do this.
* Hence we cannot just preserve the tree that was returned. So, we must copy the local structure and provide
* a function that can free the allocated space. We cannot know how to free user allocated elements and
* presume that the user will know to do this using their own factories for the structures they allocate.
*/
listLabelAST(label) ::= <<
if (list_<label> == NULL)
{
    list_<label>=ctx->vectors->newVector(ctx->vectors);
}
{
    RETURN_TYPE_<label> * tcopy;

    tcopy = (RETURN_TYPE_<label> *)ANTLR3_MALLOC(sizeof(RETURN_TYPE_<label>)); /* Note no
memory allocation checks! */
    ANTLR3_MEMCPY((void *)tcopy, (const void *)<label>, sizeof(RETURN_TYPE_<label>));
    list_<label>->add(list_<label>, (void *)tcopy, freeScope); /* Add whatever the return type is */<n>
}

```

```

>>

// R e w r i t e

rewriteCode(
  alts,
  description,
  referencedElementsDeep, // ALL referenced elements to right of ->
  referencedTokenLabels,
  referencedTokenListLabels,
  referencedRuleLabels,
  referencedRuleListLabels,
  referencedWildcardLabels,
  referencedWildcardListLabels,
  rewriteBlockLevel,
  enclosingTreeLevel,
  treeLevel) ::=
<<

/* AST REWRITE
* elements      : <referencedElementsDeep; separator=", ">
* token labels  : <referencedTokenLabels; separator=", ">
* rule labels   : <referencedRuleLabels; separator=", ">
* token list labels : <referencedTokenListLabels; separator=", ">
* rule list labels : <referencedRuleListLabels; separator=", ">
*/
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) <\n>
<endif>
{
  <rewriteCodeLabelsDecl()>
  <rewriteCodeLabelsInit()>
  root_0 = (<ASTLabelType>)(ADAPTOR->nilNode(ADAPTOR));
  <prevRuleRootRef()>.tree = root_0;
  <alts:rewriteAlt(); separator="else ">
  <if(TREE_PARSER)>
  <if(rewriteMode)>
  <prevRuleRootRef()>.tree = (<ASTLabelType>)(ADAPTOR->rulePostProcessing(ADAPTOR, root_0));
  INPUT->replaceChildren(INPUT, ADAPTOR->getParent(ADAPTOR, retval.start),
    ADAPTOR->getChildIndex(ADAPTOR, retval.start),
    ADAPTOR->getChildIndex(ADAPTOR, _last),
    retval.tree);
  <endif>
  <endif>
  <prevRuleRootRef()>.tree = root_0; // set result root
  <rewriteCodeLabelsFree()>
}

```

```

>>

rewriteCodeLabelsDecl() ::= <<
<referencedTokenLabels
  :{it | pANTLR3_REWRITE_RULE_<rewriteElementType>_STREAM stream_<it>;}
  separator="\n"
>
<referencedTokenListLabels
  :{it | pANTLR3_REWRITE_RULE_<rewriteElementType>_STREAM stream_<it>;}
  separator="\n"
>
<referencedRuleLabels
  :{it | pANTLR3_REWRITE_RULE_SUBTREE_STREAM stream_<it>;}
  separator="\n"
>
<referencedRuleListLabels
  :{it | pANTLR3_REWRITE_RULE_SUBTREE_STREAM stream_<it>;}
  separator="\n"
>
>>

rewriteCodeLabelsInit() ::= <<
<referencedTokenLabels
  :{it | stream_<it>=antlr3RewriteRule<rewriteElementType>StreamNewAEE(ADAPTOR, RECOGNIZER,
  (pANTLR3_UINT8)"token <it>", <it>);}
  separator="\n"
>
<referencedTokenListLabels
  :{it | stream_<it>=antlr3RewriteRule<rewriteElementType>StreamNewAEV(ADAPTOR, RECOGNIZER,
  (pANTLR3_UINT8)"token <it>", list_<it>);}
  separator="\n"
>
<referencedRuleLabels
  :{it | stream_<it>=antlr3RewriteRuleSubtreeStreamNewAEE(ADAPTOR, RECOGNIZER,
  (pANTLR3_UINT8)"token <it>", <it>.tree != NULL ? <it>.tree : NULL);}
  separator="\n"
>
<referencedRuleListLabels
  :{it | stream_<it>=antlr3RewriteRuleSubtreeStreamNewAEV(ADAPTOR, RECOGNIZER,
  (pANTLR3_UINT8)"token <it>", list_<it>);}
  separator="\n"
>
>>

rewriteCodeLabelsFree() ::= <<
<referencedTokenLabels
  :{it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>);}
  separator="\n"
>

```



```

<referencedTokenListLabels
:{it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>)};
separator="\n"
>
<referencedRuleLabels
:{it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>)};
separator="\n"
>
<referencedRuleListLabels
:{it | if (stream_<it> != NULL) stream_<it>->free(stream_<it>)};
separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
alt,
rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
{
if ( <referencedElementsDeep:{el | (stream_<el> != NULL && stream_<el>->hasNext(stream_<el>)) } ;
separator="|| "> )
{
<alt>
}
<referencedElementsDeep:{el | if ( stream_<el> != NULL) stream_<el>->reset(stream_<el>);<n>}>
}<n>
>>

rewriteClosureBlock(
alt,
rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
{
while ( <referencedElements:{el | (stream_<el> != NULL && stream_<el>->hasNext(stream_<el>)) } ; separator="||
"> )
{
<alt>

```

```

}
<referencedElements:{el | if (stream_<el> != NULL) stream_<el>->reset(stream_<el>);<\n>}>
}<\n>
>>
RewriteEarlyExitException() ::=
<<
CONSTRUCTEX();
EXCEPTION->type      = ANTLR3_REWRITE_EARLY_EXCEPTION;
EXCEPTION->name      = (void *)ANTLR3_REWRITE_EARLY_EXCEPTION_NAME;
>>
rewritePositiveClosureBlock(
alt,
rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
if (<referencedElements:{el | (stream_<el> == NULL || !stream_<el>->hasNext(stream_<el>)) }; separator="|| ">)
{
  <RewriteEarlyExitException()>
}
else
{
  while (<referencedElements:{el | (stream_<el>->hasNext(stream_<el>)) }; separator="|| ">) {
    <alt>
  }
  <referencedElements:{el | stream_<el>->reset(stream_<el>);<\n>}>
}
>>

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred>)
{
  <a.alt>
}<\n>
<else>
{
  <a.alt>
}<\n>
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = NULL; /* \<-- rewriteEmptyAlt() */"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<

```

```

// <fileName>:<description>
{
  <ASTLabelType> root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->nilNode(ADAPTOR));
  <root:rewriteElement()>
  <children:rewriteElement()>
  ADAPTOR->addChild(ADAPTOR, root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,terminalOptions,args) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <createRewriteNodeFromElement(...)>);<\n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<label> == NULL ? NULL : stream_<label>-
>nextNode(stream_<label>));<\n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<label> == NULL ? NULL : stream_<label>-
>nextNode(stream_<label>));<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRootToken(ADAPTOR, stream_<label> == NULL ?
NULL : stream_<label>->nextToken(stream_<label>), root_<treeLevel>));<\n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */
rewriteTokenRefRoot(token,elementIndex,terminalOptions,args) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR,
<createRewriteNodeFromElement(...)>, root_<treeLevel>));<\n>
>>

```

```

rewriteImaginaryTokenRef(args,token,terminalOptions,elementIndex) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <createImaginaryNode(tokenType=token, ...)>);<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,terminalOptions,elementIndex) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR,
<createImaginaryNode(tokenType=token, ...)>, root_<treeLevel>));<\n>
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule,dup) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<rule> == NULL ? NULL : stream_<rule>-
>nextTree(stream_<rule>));<\n>
>>

rewriteRuleRefRoot(rule,dup) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, stream_<rule> == NULL ? NULL
: stream_<rule>->nextNode(stream_<rule>), root_<treeLevel>));<\n>
>>

rewriteNodeAction(action) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, <action>);<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = (<ASLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, <action>, root_<treeLevel>));<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<label> == NULL ? NULL : stream_<label>-
>nextTree(stream_<label>));<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<label> == NULL ? NULL : stream_<label>-

```

```

>nextTree(stream_<label>));<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRoot(ADAPTOR, stream_<label> == NULL ? NULL
: stream_<label>->nextNode(stream_<label>), root_<treeLevel>));<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = (<ASTLabelType>)(ADAPTOR->becomeRoot((<ASTLabelType>)(stream_<label> == NULL
? NULL : stream_<label>->nextNode(stream_<label>), root_<treeLevel>));<\n>
>>

rewriteWildcardLabelRef(label) ::= <<
ADAPTOR->addChild(ADAPTOR, root_<treeLevel>, stream_<label> == NULL ? NULL : stream_<label>-
>nextTree(stream_<label>));<\n>
>>

createImaginaryNode(tokenType,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
<! new MethodNode(IDLabel, args) !>
<terminalOptions.node>New(<tokenType><if(args)>, <args; separator=","><endif>)
<else>
<if(args)>

#if <length(args)> == 2
(<ASTLabelType>)ADAPTOR->createTypeTokenText(ADAPTOR, <tokenType>, TOKTEXT(<args; separator=",">,
">))
#else
(<ASTLabelType>)ADAPTOR->createTypeText(ADAPTOR, <tokenType>, (pANTLR3_UINT8)<args;
separator=","> ">)
#endif

<else>
(<ASTLabelType>)ADAPTOR->createTypeText(ADAPTOR, <tokenType>, (pANTLR3_UINT8)"<tokenType>")
<endif>
<endif>
>>

createRewriteNodeFromElement(token,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
<terminalOptions.node>New(stream_<token>->nextToken(stream_<token>)<if(args)>, <args; separator=",">,
"><endif>)
<else>
<if(args)> <! must create new node from old !>

```

```

#if <length(args)> == 2
ADAPTOR->createTypeTokenText(ADAPTOR, <token>->getType(<token>, TOKTEXT(<token>, <args;
separator=", ">)) /* JIMI */
#else
ADAPTOR->createTypeToken(ADAPTOR, <token>->getType(<token>, <token>, <args; separator=", ">)
#endif

<else>
stream_<token> == NULL ? NULL : stream_<token>->nextNode(stream_<token>)
<endif>
<endif>
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/C/AST.stg

No license file was found, but licenses were detected in source scan.

```

/*
 * [The "BSD license"]
 * Copyright (c) 2007-2008 Johannes Luber
 * Copyright (c) 2005-2007 Kunle Odutola
 * Copyright (c) 2011 Sam Harwell
 * Copyright (c) 2011 Terence Parr
 * All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in the
 * documentation and/or other materials provided with the distribution.
 * 3. The name of the author may not be used to endorse or promote products
 * derived from this software without specific prior written permission.
 *
 * THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
 * IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
 * IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
 * INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
 * DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
 * THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
 * (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
 * THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
 */
/** Template subgroup to add template rewrite output

```

```

* If debugging, then you'll also get STDbg.stg loaded.
*/

@outputFile.imports() ::= <<
<@super.imports()>
using Antlr.StringTemplate;
using Antlr.StringTemplate.Language;
<if(!backtracking)>
using Hashtable = System.Collections.Hashtable;
<endif>

>>

/** Add this to each rule's return value struct */
@returnScope.ruleReturnMembers() ::= <<
private StringTemplate _st;
public StringTemplate Template { get { return _st; } set { _st = value; } }
public override string ToString() { return (Template==null) ? string.Empty : Template.ToString(); }
>>

@genericParser.members() ::= <<
<@super.members()>
protected StringTemplateGroup templateLib = new StringTemplateGroup("<name>Templates",
typeof(AngleBracketTemplateLexer) );

public StringTemplateGroup TemplateLib
{
    get { return this.templateLib; }
    set { this.templateLib = value; }
}

/// \<summary> Allows convenient multi-value initialization:
/// "new STAttrMap().Add(...).Add(...)"
/// \</summary>
protected class STAttrMap : Hashtable
{
    public STAttrMap Add(string attrName, object value)
    {
        base.Add(attrName, value);
        return this;
    }
    public STAttrMap Add(string attrName, int value)
    {
        base.Add(attrName, value);
        return this;
    }
}
>>

```

```

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label+".Template",...)>
>>

```

```

rewriteTemplate(alts) ::= <<

```

```

// TEMPLATE REWRITE
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> )
{
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
}
<else>
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>

```

```

replaceTextInLine() ::= <<
<if(TREE_PARSER)>
((TokenRewriteStream)input.TokenStream).Replace(
input.TreeAdaptor.GetTokenStartIndex(retval.Start),
input.TreeAdaptor.GetTokenStopIndex(retval.Start),
retval.Template);
<else>
((TokenRewriteStream)input).Replace(
((IToken)retval.Start).TokenIndex,
input.LT(-1).TokenIndex,
retval.Template);
<endif>
>>

```

```

rewriteTemplateAlt() ::= <<
// <it.description>
<if(it.pred)>
if (<it.pred> ) {
retval.Template = <it.alt>;
}<\n>
<else>
{
retval.Template = <it.alt>;
}<\n>
<endif>
>>

```



```

rewriteEmptyTemplate(alts) ::= <<
null;
>>

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template after having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded
 * template.
 */
rewriteExternalTemplate(name,args) ::= <<
templateLib.GetInstanceOf("<name>"<if(args)>,
new STAttrMap()<args:{a | .Add("<a.name>", <a.value>)}>
<endif>)
>>

/** expr is a string expression that says what template to load */
rewriteIndirectTemplate(expr,args) ::= <<
templateLib.GetInstanceOf(<expr><if(args)>,
new STAttrMap()<args:{a | .Add("<a.name>", <a.value>)}>
<endif>)
>>

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <<
new StringTemplate(templateLib, "<template>"<if(args)>,
new STAttrMap()<args:{a | .Add("<a.name>", <a.value>)}>
<endif>)
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
<action>
>>

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
(<st>).SetAttribute("<attrName>",<expr>);
>>

/** Translate % {stringExpr} */
actionStringConstructor(stringExpr) ::= <<
new StringTemplate(templateLib,<stringExpr>)
>>

```

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-

jar/org/antlr/codegen/templates/CSharp2/ST.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2007-2008 Johannes Luber

* Copyright (c) 2005-2007 Kunle Odutola

* Copyright (c) 2011 Sam Harwell

* Copyright (c) 2011 Terence Parr

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during tree parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* Each combination has its own template except that label/no label

* is combined into tokenRef, ruleRef, ...

*/

/** Add a variable to track last element matched */

ruleDeclarations() ::= <<


```

    <children:element()>
    Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
  }
<else>
Match(input, TokenType.Down, null); <checkRuleBacktrackFailure()>
<children:element()>
Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
<endif>
<if(!rewriteMode)>
adaptor.AddChild(root_<enclosingTreeLevel>, root_<treeLevel>);
<endif>
_last = _save_last_<treeLevel>;
}
<else>
<super.tree(...)>
<endif>
>>

// TOKEN AST STUFF

/** ID! and output=AST (same as plain tokenRef) 'cept add
 * setting of _last
 */
tokenRefBang(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<else>
<super.tokenRefBang(...)>
<endif>
>>

/** ID auto construct */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);
<endif><\n>

```

```

adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>;
<endif>
<else>
<super.tokenRef(...)>
<endif>
>>

/** label+=TOKEN auto construct */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
<tokenRef(...)>
<listLabelElem(elem=label,...)>
<else>
<super.tokenRefAndListLabel(...)>
<endif>
>>

/** ^(ID ...) auto construct */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.tokenRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node><(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(if(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}<endif>
<endif>
<else>
<super.tokenRefRuleRoot(...)>
<endif>
>>

/** Match ^(label+=TOKEN ...) auto construct */
tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>

```

```

<tokenRefRuleRoot(...)>
<listLabelElem(elem=label,...)>
<else>
<super.tokenRefRuleRootAndListLabel(...)>
<endif>
>>

/** Match . wildcard and auto dup the node/subtree */
wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.wildcard(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.DupTree(<label>);
adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}<endif>
<else> <! rewrite mode !>
<if(backtracking)>if (<actions.(actionScope).synpredgate>)<endif>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>;
<endif>
<else>
<super.wildcard(...)>
<endif>
>>

// SET AST

matchSet(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<if(terminalOptions.type)><terminalOptions.type>,<endif><label><if(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);
<endif><\n>
adaptor.AddChild(root_<treeLevel>, <label>_tree);
<if(backtracking)>}\}<endif>
<endif>
}, ...
)>

```

```

<else>
<super.matchSet(...)>
<endif>
>>

matchRuleBlockSet(s,label,terminalOptions,elementIndex,postmatchCode,treeLevel="0") ::= <<
<if(!ruleDescriptor.isSynPred)>
<matchSet(...)>
<noRewrite(...)> <! set return tree !>
<else>
<super.matchRuleBlockSet(...)>
<endif>
>>

matchSetBang(s,label,terminalOptions,elementIndex,postmatchCode) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.matchSet(...)>
<else>
<super.matchSetBang(...)>
<endif>
>>

matchSetRuleRoot(s,label,terminalOptions,elementIndex,debug) ::= <<
<if(!ruleDescriptor.isSynPred)>
<super.matchSet(postmatchCode={
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<if(terminalOptions.node)>
<label>_tree = new
<terminalOptions.node><(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(if(terminalOptions.tex
t)>,<terminalOptions.text; format="string"><endif>);
<else>
<label>_tree =
(<ASTLabelType>)adaptor.DupNode(<(if(terminalOptions.type)><terminalOptions.type>,<endif><label><(termin
alOptions.text)>,<terminalOptions.text; format="string"><endif>);
<endif><\n>
root_<treeLevel> = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_<treeLevel>);
<if(backtracking)>}\}<endif>
<endif>
}, ...
)>
<else>
<super.matchSetRuleRoot(...)>
<endif>
>>

// RULE REF AST

```

```

/** rule auto construct */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate><endif>
<if(!rewriteMode)>
adaptor.AddChild(root_<treeLevel>, <label>.Tree);
<else> <! rewrite mode !>
if (_first_<treeLevel> == null) _first_<treeLevel> = <label>.Tree;
<endif>
<else>
<super.ruleRef(...)>
<endif>
>>

/** x+=rule auto construct */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<ruleRef(...)>
<listLabelElem(elem={<label>.Tree},...)>
<else>
<super.ruleRefAndListLabel(...)>
<endif>
>>

/** ^(rule ...) auto construct */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRef(...)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate> <endif>root_<treeLevel> =
(<ASTLabelType>)adaptor.BecomeRoot(<label>.Tree, root_<treeLevel>);
<endif>
<else>
<super.ruleRefRuleRoot(...)>
<endif>
>>

/** ^(x+=rule ...) auto construct */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<ruleRefRuleRoot(...)>
<listLabelElem(elem={<label>.Tree},...)>
<else>
<super.ruleRefRuleRootAndListLabel(...)>

```



```

<endif>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrack(...)>
<else>
<super.ruleRefTrack(...)>
<endif>
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefTrackAndListLabel(...)>
<else>
<super.ruleRefTrackAndListLabel(...)>
<endif>
>>

/** ^(rule ...) rewrite */
ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRuleRootTrack(...)>
<else>
<super.ruleRefRuleRootTrack(...)>
<endif>
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
_last = (<ASTLabelType>)input.LT(1);
<super.ruleRefRuleRootTrackAndListLabel(...)>
<else>
<super.ruleRefRuleRootTrackAndListLabel(...)>
<endif>
>>

/** Streams for token refs are tree nodes now; override to
 * change NextToken to NextNode.
 */
createRewriteNodeFromElement(token,terminalOptions,args) ::= <%
<if(terminalOptions.node)>

```

```
new
<terminalOptions.node>(<if(terminalOptions.type)><terminalOptions.type>,<endif>stream_<token>.NextNode())
<else>
stream_<token>.NextNode()
<endif>
%>
```

```
ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(!ruleDescriptor.isSynPred)>
<if(!rewriteMode)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
retval.Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
<if(backtracking)>}<endif>
<endif>
<endif>
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp2/ASTTreeParser.stg
No license file was found, but licenses were detected in source scan.
```

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2012 Terence Parr
```

```
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** The API version of the runtime that recognizers generated by this runtime
* need.

*/

apiVersion() ::= "1"

/** The overall file structure of a recognizer; stores methods for rules
* and cyclic DFAs plus support code.

*/

outputFile(LEXER,PARSER,TREE_PARSER, actionScope, actions,
docComment, recognizer,
name, tokens, tokenNames, rules, cyclicDFAs,
bitsets, buildTemplate, buildAST, rewriteMode, profile,
backtracking, synpreds, memoize, numRules,
fileName, ANTLRVersion, generatedTimestamp, trace,
scopes, superClass, literals) ::=

<<

\$ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>

<@imports>

import sys

from antlr3 import *

<if(TREE_PARSER)>

from antlr3.tree import *<\n

<endif>

<@end>

<actions.(actionScope).header>

<! <docComment> !>

for convenience in actions

HIDDEN = BaseRecognizer.HIDDEN

token types

<tokens:{it | <it.name>=<it.type>}; separator="\n">

token names

tokenNamesMap = {

0: "\<invalid>", 1: "\<EOR>", 2: "\<DOWN>", 3: "\<UP>",

<tokens:{it | <it.type>: "<it.name>"}; wrap, separator=", ">

}

Token.registerTokenNamesMap(tokenNamesMap)

<recognizer>

```

<if(actions.(actionScope).main)>
<actions.(actionScope).main>
<else>
def main(argv, stdin=sys.stdin, stdout=sys.stdout, stderr=sys.stderr):
<if(LEXER)>
    from antlr3.main import LexerMain
    main = LexerMain(<recognizer.name>)<\n>
<endif>
<if(PARSER)>
    from antlr3.main import ParserMain
    main = ParserMain("<recognizer.grammar.name>Lexer", <recognizer.name>)<\n>
<endif>
<if(TREE_PARSER)>
    from antlr3.main import WalkerMain
    main = WalkerMain(<recognizer.name>)<\n>
<endif>
    main.stdin = stdin
    main.stdout = stdout
    main.stderr = stderr
    main.execute(argv)<\n>
<endif>

<actions.(actionScope).footer>

if __name__ == '__main__':
    main(sys.argv)

>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode,
    labelType="CommonToken", superClass="Lexer") ::= <<
<if(grammar.directDelegates)>
# path hack to allow absolute import of related grammars.
from os.path import dirname
__path__ = [dirname(__file__)]
del dirname

<grammar.directDelegates:
{g|from .<g.recognizerName> import <g.recognizerName>}; separator="\n">
<endif>

class <grammar.recognizerName>(<@superClassName><superClass><@end>):
    <scopes: {it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>

    grammarFileName = "<fileName>"
    api_version = <apiVersion()>

    def __init__(self<grammar.delegators: {g|, <g:delegateName()>}>, input=None, state=None):

```

```

    if state is None:
        state = RecognizerSharedState()
    super().__init__(input, state)

<if(memoize)>
<if(grammar.grammarIsRoot)>
    self._state.ruleMemo = {}
<endif>
<endif>

    <grammar.directDelegates:
        {g|self.<g:delegateName()> = <g.recognizerName>(<trunc(g.delegators):{p|<p:delegateName()>, }>self, input,
state)}; separator="\n">
    <grammar.directDelegates:
        {g|<g.delegates:{h|self.<h:delegateName()> = self.<g:delegateName()>.<h:delegateName()>};
separator="\n">}; separator="\n">
    <grammar.delegators:
        {g|self.<g:delegateName()> = <g:delegateName()>}; separator="\n">
    <last(grammar.delegators):
        {g|self.gParent = <g:delegateName()>}; separator="\n">
    self.delegates = [<grammar.delegates: {g|self.<g:delegateName()>}; separator = ", ">]

    <cyclicDFAs:{ dfa | <cyclicDFAInit(dfa)>}; separator="\n">

    <actions.lexer.init>

    <actions.lexer.members>

<if(filterMode)>
    <filteringNextToken()>
<endif>
    <rules; separator="\n\n">

    <synpreds:{ p | <lexerSynpred(p)>}>

    <cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

>>

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */

```

```

filteringNextToken() ::= <<
def nextToken(self):
    while True:
        if self.input.LA(1) == EOF:
            return self.makeEOFToken()

        self._state.token = None
        self._state.channel = DEFAULT_CHANNEL
        self._state.tokenStartCharIndex = self.input.index()
        self._state.tokenStartCharPositionInLine = self.input.charPositionInLine
        self._state.tokenStartLine = self.input.line
        self._state._text = None
        try:
            m = self.input.mark()
            try:
                # means we won't throw slow exception
                self._state.backtracking = 1
                try:
                    self.mTokens()
                finally:
                    self._state.backtracking = 0

            except BacktrackingFailed:
                # mTokens backtracks with synpred at backtracking==2
                # and we set the synpredgate to allow actions at level 1.
                self.input.rewind(m)
                self.input.consume() # advance one char and try again

        else:
            self.emit()
            return self._state.token

    except RecognitionException as re:
        # shouldn't happen in backtracking mode, but...
        self.reportError(re)
        self.recover(re)

def memoize(self, input, ruleIndex, ruleStartIndex, success):
    if self._state.backtracking > 1:
        # is Lexer always superclass?
        super().memoize(input, ruleIndex, ruleStartIndex, success)

def alreadyParsedRule(self, input, ruleIndex):
    if self._state.backtracking > 1:
        return super().alreadyParsedRule(input, ruleIndex)
    return False

```

```

>>

actionGate() ::= "self._state.backtracking == 0"

filteringActionGate() ::= "self._state.backtracking == 1"

/** How to generate a parser */

genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
              bitsets, inputStreamType, superClass, labelType, members,
              rewriteElementType, filterMode, init, ASTLabelType="Object") ::= <<
# token names
tokenNames = [
  "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>",
  <tokenNames; wrap, separator=", ">
]

<scopes:{it|<if(it.isDynamicGlobalScope)><globalAttributeScopeClass(scope=it)><endif>>>

<if(grammar.directDelegates)>
# path hack to allow absolute import of related grammars.
from os.path import dirname
__path__ = [dirname(__file__)]
del dirname

<grammar.directDelegates:
{g|from .<g.recognizerName> import <g.recognizerName>}; separator="\n">
<endif>

<rules:{it|<ruleAttributeScopeClass(scope=it.ruleDescriptor.ruleScope)>>>

class <grammar.recognizerName>(<@superClassName><superClass><@end>):
  grammarFileName = "<fileName>"
  api_version = <apiVersion()>
  tokenNames = tokenNames

  def __init__(self<grammar.delegators:{g|, <g.delegateName()>>, input, state=None, *args, **kwargs):
    if state is None:
      state = RecognizerSharedState()

    <@args()>
    super().__init__(input, state, *args, **kwargs)

<if(memoize)>
<if(grammar.grammarIsRoot)>
  self._state.ruleMemo = {}

```

```

<endif>
<endif>

<cyclicDFAs:{ dfa | <cyclicDFAInit(dfa)>}; separator="\n">

<scopes:{ it | <if(it.isDynamicGlobalScope)><globalAttributeScopeStack(scope=it)><endif>}>
<rules:{ it | <ruleAttributeScopeStack(scope=it.ruleDescriptor.ruleScope)>}>

<init>

<grammar.delegators:
  {g|self.<g.delegateName()> = <g.delegateName()>}; separator="\n">
<grammar.directDelegates:
  {g|self.<g.delegateName()> = <g.recognizerName>(<trunc(g.delegators):{p|<p.delegateName()>, }>self, input,
state)); separator="\n">
<grammar.directDelegates:
  {g|<g.delegates:{h|self.<h.delegateName()> = self.<g.delegateName()>.<h.delegateName()>};
separator="\n">}; separator="\n">
<last(grammar.delegators):
  {g|self.gParent = self.<g.delegateName()>}; separator="\n">
self.delegates = [<grammar.delegates: {g|self.<g.delegateName()>}; separator = ", ">]

<@init><@end>

<@members><@end>

<members>

<rules; separator="\n\n">

<! generate rule/method definitions for imported rules so they
  appear to be defined in this recognizer. !>
<grammar.delegatedRules:{ ruleDescriptor| <delegateRule(ruleDescriptor)> }; separator="\n">

<synpreds:{ p | <synpred(p)>}>

<cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>

<bitsets:{ it | FOLLOW_<it.name>_in_<it.inName><it.tokenIndex> = frozenset([<it.tokenTypes:{ it |
<it>};separator=", ">])<n>}>

>>

delegateRule(ruleDescriptor) ::= <<
def <ruleDescriptor.name>(self, <ruleDescriptor.parameterScope:parameterScope()>):
<\ > <if(ruleDescriptor.hasReturnValue)>return
<endif>self.<ruleDescriptor.grammar:delegateName()>.<ruleDescriptor.name>(<if(ruleDescriptor.parameterScope)

```



```

><ruleDescriptor.parameterScope.attributes:{a|<a.name>}; separator=", "><endif>

>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,
    ASTLabelType="Object", superClass="Parser", labelType="Token",
    members={<actions.parser.members>},
    init={<actions.parser.init>}
) ::= <<
<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, "TokenStream", superClass,
    labelType, members, "Token",
    false, init, ASTLabelType)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
    numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="Object",
superClass={<if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif>},
members={<actions.treeparser.members>},
init={<actions.treeparser.init>}
) ::= <<
<genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, "TreeNodeStream", superClass,
    labelType, members, "Node",
    filterMode, init, ASTLabelType)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
# $ANTLR start "<ruleName>"
def <ruleName>_fragment(self, <ruleDescriptor.parameterScope:parameterScope()):
    <ruleLabelDefs(>
<if(trace)>
    self.traceIn("<ruleName>_fragment", <ruleDescriptor.index>)
    try:
        <block>

```

```

    finally:
        self.traceOut("<ruleName>_fragment", <ruleDescriptor.index>)

<else>
    <block>
<endif>
# $ANTLR end "<ruleName>"

>>

synpred(name) ::= <<
def <name>(self):
    self._state.backtracking += 1
    <@start()>
    start = self.input.mark()
    try:
        self.<name>_fragment()
    except BacktrackingFailed:
        success = False
    else:
        success = True
    self.input.rewind(start)
    <@stop()>
    self._state.backtracking -= 1
    return success

>>

lexerSynpred(name) ::= <<
<synpred(name)>
>>

ruleMemoization(name) ::= <<
<if(memoize)>
if self._state.backtracking > 0 and self.alreadyParsedRule(self.input, <ruleDescriptor.index>):
    # for cached failed rules, alreadyParsedRule will raise an exception
    success = True
    return <ruleReturnValue()>

<endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>
if self._state.backtracking > 0:

```

```

    raise BacktrackingFailed

<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<returnScope(scope=ruleDescriptor.returnScope)>

# $ANTLR start "<ruleName>"
# <fileName>:<description>
<ruleDescriptor.actions.decorate>
def <ruleName>(self, <ruleDescriptor.parameterScope:parameterScope()):
<if(trace)>
    self.traceIn("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
    <ruleLabelDefs()>
    <ruleDescriptor.actions.init>
    <@preamble()>
    <@body><ruleBody()><@end>
    <@postamble()>
    return <ruleReturnValue()>

# $ANTLR end "<ruleName>"
>>

ruleBody() ::= <<
<if(memoize)>
<if(backtracking)>
success = False<\n>
<endif>
<endif>
try:
    try:
        <ruleMemoization(name=ruleName)>
        <block>
        <ruleCleanUp()>
        <(ruleDescriptor.actions.after):execAction()>

<if(memoize)>
<if(backtracking)>
    success = True<\n>
<endif>
<endif>

```

```

<if(exceptions)>
  <exceptions:{e|<catch(decl=e.decl,action=e.action)><\n>}>
<else>
<if(!emptyRule)>
<if(actions.(actionScope).rulecatch)>
  <actions.(actionScope).rulecatch>
<else>
  except RecognitionException as re:
    self.reportError(re)
    self.recover(self.input, re)
    <@setErrorReturnValue()>

<endif>
<else>
  finally:
    pass

<endif>
<endif>
finally:
<if(trace)>
  self.traceOut("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
  <memoize()>
  <ruleScopeCleanUp()>
  <finally>
  pass
>>

catch(decl,action) ::= <<
except <e.decl>:
  <e.action>

>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval = self.<ruleDescriptor.name>_return()
retval.start = self.input.LT(1)<\n>
<elseif(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes:{ a |
<a.name> = <if(a.initValue)><a.initValue><else>None<endif>
}>
<endif>
<if(memoize)>
<ruleDescriptor.name>_startIndex = self.input.index()
<endif>
>>

```

```

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes:{it | self.<it>_stack.append(<it>_scope()); separator="\n">
<ruleDescriptor.ruleScope:{it | self.<it.name>_stack.append(<it.name>_scope()); separator="\n">
>>

```

```

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes:{it | self.<it>_stack.pop(); separator="\n">
<ruleDescriptor.ruleScope:{it | self.<it.name>_stack.pop(); separator="\n">
>>

```

```

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,
ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wildcardTreeListLabels]
: {it | <it.label.text> = None }; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels,
ruleDescriptor.wildcardTreeListLabels]
: {it | list_<it.label.text> = None }; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels:{it | <it.label.text> = None }; separator="\n">
>>

```

```

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it | <it.label.text> = None }; separator="\n"
>
<ruleDescriptor.charLabels:{it | <it.label.text> = None }; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels]
: {it | list_<it.label.text> = None }; separator="\n"
>
>>

```

```

ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>
<endif>
%>

```

```

ruleCleanup() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.stop = self.input.LT(-1)<\n>
<endif>
<endif>
>>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if self._state.backtracking > 0:
    self.memoize(self.input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex, success)

<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
# $ANTLR start "<ruleName>"
def m<ruleName>(self, <ruleDescriptor.parameterScope:parameterScope(>):
<if(trace)>
    self.traceIn("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
    <ruleScopeSetUp()>
    <ruleDeclarations()>
<if(memoize)>
<if(backtracking)>
    success = False<\n>
<endif>
<endif>
    try:
<if(nakedBlock)>
        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>
        <block><\n>
<else>
        _type = <ruleName>
        _channel = DEFAULT_CHANNEL

        <ruleMemoization(name=ruleName)>
        <lexerRuleLabelDefs()>
        <ruleDescriptor.actions.init>

```

```

    <block>
    <ruleCleanUp()>
    self._state.type = _type
    self._state.channel = _channel
    <(ruleDescriptor.actions.after):execAction()>
<endif>
<if(memoize)>
<if(backtracking)>
    success = True<\n>
<endif>
<endif>

    finally:
<if(trace)>
    self.traceOut("<ruleName>", <ruleDescriptor.index>)<\n>
<endif>
    <ruleScopeCleanUp()>
    <memoize()>
    pass

# $ANTLR end "<ruleName>"

>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
def mTokens(self):
    <block><\n>

>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
alt<decisionNumber> = <maxAlt>
<decls>
<@body><blockBody()><@end>
>>

blockBody() ::= <<
<@predecision()>
<@decision><decision><@end>

```

```

<@postdecision(>
<@prebranch(>
<alts:{ a | <altSwitchCase(i, a)> }; separator="\nel">
<@postbranch(>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
alt<decisionNumber> = <maxAlt>
<decls>
<@predecision(>
<@decision><decision><@end>
<@postdecision(>
<alts:{ a | <altSwitchCase(i, a)> }; separator="\nel">
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
# <fileName>:<description>
<decls>
<@prealt(>
<alts>
<@postalt(>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
# <fileName>:<description>
cnt<decisionNumber> = 0
<decls>
<@preloop(>
<@loopBody>
<positiveClosureBlockLoop(>
<@end>
<@postloop(>
>>

positiveClosureBlockLoop() ::= <<

```



```

while True: #loop<decisionNumber>
  alt<decisionNumber> = <maxAlt>
  <@predecision()>
  <@decisionBody><decision><@end>
  <@postdecision()>
  <alts:{a | <altSwitchCase(i, a)>}; separator="\nel">
  else:
    if cnt<decisionNumber> >= 1:
      break #loop<decisionNumber>

  <ruleBacktrackFailure()>
  eee = EarlyExitException(<decisionNumber>, self.input)
  <@earlyExitException()>
  raise eee

  cnt<decisionNumber> += 1
>>

positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
# <fileName>:<description>
<decls>
<@preloop()>
<@loopBody>
<closureBlockLoop()>
<@end>
<@postloop()>
>>

closureBlockLoop() ::= <<
while True: #loop<decisionNumber>
  alt<decisionNumber> = <maxAlt>
  <@predecision()>
  <@decisionBody><decision><@end>
  <@postdecision()>
  <alts:{a | <altSwitchCase(i, a)>}; separator="\nel">
  else:
    break #loop<decisionNumber>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation
* so we can just use the normal block template
*/

```

```
optionalBlock ::= block
```

```
optionalBlockSingleAlt ::= block
```

```
/** A case in a switch that jumps to an alternative given the alternative  
 * number. A DFA predicts the alternative and then a simple switch  
 * does the jump to the code that actually matches that alternative.  
 */
```

```
altSwitchCase(altNum,alt) ::= <<  
if alt<decisionNumber> == <altNum>:  
  <@prealt()>  
  <alt>  
>>
```

```
/** An alternative is just a list of elements; at outermost level */  
alt(elements,altNum,description,autoAST,outerAlt, treeLevel,rew) ::= <<  
# <fileName>:<description>  
pass <! so empty alternatives are a valid block !>  
<@declarations()>  
<elements:element()>  
<rew>  
<@cleanup()>  
>>
```

```
/** What to emit when there is no rewrite. For auto build  
 * mode, does nothing.  
 */  
noRewrite(rewriteBlockLevel, treeLevel) ::= ""
```

```
// E L E M E N T S
```

```
/** Dump the elements one per line */  
element(e) ::= <<  
<@prematch()>  
<e.el><\n>  
>>
```

```
/** match a token optionally with a label in front */  
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<  
<if(label)><label> = <endif>self.match(self.input, <token>,  
self.FOLLOW_<token>_in_<ruleName><elementIndex>  
>>
```

```
/** ids+=ID */  
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<  
<tokenRef(token,label,elementIndex,terminalOptions)>  
<listLabel(label, label)>  
>>
```

```

listLabel(label, elem) ::= <<
if list_<label> is None:
    list_<label> = []
list_<label>.append(<elem>)<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = self.input.LA(1)<\n>
<endif>
self.match(<char>)
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = self.input.LA(1)<\n>
<endif>
self.matchRange(<a>, <b>)
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="",terminalOptions={}) ::= <<
<if(label)>
<label> = self.input.LT(1)<\n>
<endif>
if <s>:
    self.input.consume()
    <postmatchCode>
<if(!LEXER)>
    self._state.errorRecovery = False<\n>
<endif>

else:
    <ruleBacktrackFailure()>
    mse = MismatchedSetException(None, self.input)
    <@mismatchedSetException()>
<if(LEXER)>
    self.recover(mse)
    raise mse
<else>
    raise mse
<! use following code to make it recover inline; remove throw mse;
self.recoverFromMismatchedSet(
    self.input, mse, self.FOLLOW_set_in_<ruleName><elementIndex>
)

```

```

!>
<endif>
<\n>
>>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(label, label)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex="0") ::= <<
<if(label)>
<label>Start = self.getCharIndex()
self.match(<string>)
<label>StartLine<elementIndex> = self.getLine()
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
<label> = <labelType>(input=self.input, type=INVALID_TOKEN_TYPE, channel=DEFAULT_CHANNEL,
start=<label>Start, stop=self.getCharIndex()-1)
<label>.line = <label>StartLine<elementIndex>
<label>.charPositionInLine = <label>StartCharPos<elementIndex>
<else>
self.match(<string>)
<endif>
>>

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<if(label)>
<label> = self.input.LT(1)<\n>
<endif>
self.matchAny()
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<wildcard(...)>
<listLabel(label,label)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = self.input.LA(1)<\n>
<endif>
self.matchAny()
>>

```

```

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(label, elementIndex)>
<listLabel(label, label)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
self._state.following.append(self.FOLLOW_<rule.name>_in_<ruleName><elementIndex>)
<if(label)><label> = <endif>self.<if(scope)><scope:delegateName()>.<endif><rule.name><(args; separator=",
"><\n>
self._state.following.pop()
>>

/** ids+=rule */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(rule,label,elementIndex,args,scope)>
<listLabel(label, label)>
>>

/** A lexer rule reference
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.
 */
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
<if(label)>
<label>Start<elementIndex> = self.getCharIndex()
self.<if(scope)><scope:delegateName()>.<endif>m<rule.name><(args; separator=", ">)
<label>StartLine<elementIndex> = self.getLine()
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
<label> = <labelType>(
  input=self.input,
  type=INVALID_TOKEN_TYPE,
  channel=DEFAULT_CHANNEL,
  start=<label>Start<elementIndex>,
  stop=self.getCharIndex()-1)
<label>.line = <label>StartLine<elementIndex>
<label>.charPositionInLine = <label>StartCharPos<elementIndex>
<else>
self.<if(scope)><scope:delegateName()>.<endif>m<rule.name><(args; separator=", ">)
<endif>
>>

```

```

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(rule,label,args,elementIndex,scope)>
<listLabel(label, label)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
<label>Start<elementIndex> = self.getCharIndex()
<label>StartLine<elementIndex> = self.getLine()
<label>StartCharPos<elementIndex> = self.getCharPositionInLine()
self.match(EOF)
<label> = <labelType>(input=self.input, type=EOF, channel=DEFAULT_CHANNEL,
start=<label>Start<elementIndex>, stop=self.getCharIndex()-1)
<label>.line = <label>StartLine<elementIndex>
<label>.charPositionInLine = <label>StartCharPos<elementIndex>
<else>
self.match(EOF)
<endif>
>>

// used for left-recursive rules
recRuleDefArg()          ::= "<recRuleArg()>"
recRuleArg()             ::= "_p"
recRuleAltPredicate(ruleName, opPrec) ::= "<recRuleArg()> \<= <opPrec>"
recRuleSetResultAction() ::= "root_0 = $<ruleName>_primary.tree"
recRuleSetReturnAction(src, name)    ::= "$<name> = $<src>.<name>"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if self.input.LA(1) == DOWN:
self.match(self.input, DOWN, None)
<children:element()>
self.match(self.input, UP, None)

<else>
self.match(self.input, DOWN, None)
<children:element()>
self.match(self.input, UP, None)
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is

```

```

* also hoisted into a prediction expression).
*/
validateSemanticPredicate(pred,description) ::= <<
if not (<evalPredicate(pred, description)>):
  <ruleBacktrackFailure()>
  raise FailedPredicateException(self.input, "<ruleName>", "<description>")

>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n" >
else:
<if(eotPredictsAlt)>
  alt<decisionNumber> = <eotPredictsAlt>
<else>
  <ruleBacktrackFailure()>
  nvae = NoViableAltException("<description>", <decisionNumber>, <stateNumber>, self.input)<\n>
  <@noViableAltException()>
  raise nvae<\n>
<endif>
>>

/** Same as a normal DFA state except that we don't examine lookahead
* for the bypass alternative. It delays error detection but this
* is faster, smaller, and more what people expect. For (X)? people
* expect "if ( LA(1)==X ) match(X);" and that's it.
*/
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n" >
>>

/** A DFA state that is actually the loopback decision of a closure
* loop. If end-of-token (EOT) predicts any of the targets then it
* should act like a default clause (i.e., no error can be generated).
* This is used only in the lexer so that for ('a')* on the end of a rule
* anything other than 'a' predicts exiting.
*/
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = self.input.LA(<k>)<\n>
<edges; separator="\n"><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber> = <eotPredictsAlt> <! if no edges, don't gen ELSE !>
<else>

```

```

else:
    alt<decisionNumber> = <eotPredictsAlt>
<\n>
<endif>
<endif>
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber> = <alt>"

/** A simple edge with an expression. If the expression is satisfied,
 * enter to the target state. To handle gated productions, we may
 * have to evaluate some predicates for this edge.
 */
dfaEdge(labelExpr, targetState, predicates) ::= <<
if (<labelExpr>) <if(predicates)>and (<predicates>)<endif>:
    <targetState>
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
 * decides if this is possible: CodeGenerator.canGenerateSwitch().
 */
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
<!
    FIXME: this is one of the few occasion, where I miss a switch statement
    in Python. ATM this is implemented as a list of if .. elif ..
    This may be replaced by faster a dictionary lookup, when I find a solution
    for the cases when an edge is not a plain dfaAcceptState.
!>
LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n">
else:
<if(eotPredictsAlt)>
    alt<decisionNumber> = <eotPredictsAlt>
<else>
    <ruleBacktrackFailure()>
    nvae = NoViableAltException("<description>", <decisionNumber>, <stateNumber>, self.input)<\n>
    <@noViableAltException()>
    raise nvae<\n>
<endif>

>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n">

```



```

>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
LA<decisionNumber> = self.input.LA(<k>)
<edges; separator="\n">
<if(eotPredictsAlt)>
else:
    alt<decisionNumber> = <eotPredictsAlt>
<endif>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
if LA<decisionNumber> in {<labels; separator=",">}:
    <targetState>
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = self.dfa<decisionNumber>.predict(self.input)
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
# lookup tables for DFA #<dfa.decisionNumber>

DFA<dfa.decisionNumber>_eot = DFA.unpack(
    "<dfa.javaCompressedEOT; wrap="\n \>"
)

DFA<dfa.decisionNumber>_eof = DFA.unpack(
    "<dfa.javaCompressedEOF; wrap="\n \>"
)

DFA<dfa.decisionNumber>_min = DFA.unpack(
    "<dfa.javaCompressedMin; wrap="\n \>"
)

```

```

DFA<dfa.decisionNumber>_max = DFA.unpack(
  "<dfa.javaCompressedMax; wrap=\"\\n \\\">"
)

DFA<dfa.decisionNumber>_accept = DFA.unpack(
  "<dfa.javaCompressedAccept; wrap=\"\\n \\\">"
)

DFA<dfa.decisionNumber>_special = DFA.unpack(
  "<dfa.javaCompressedSpecial; wrap=\"\\n \\\">"
)

DFA<dfa.decisionNumber>_transition = [
  <dfa.javaCompressedTransition:{s|DFA.unpack("<s; wrap=\"\\n\\\">"); separator=",\n">
]

# class definition for DFA #<dfa.decisionNumber>

class DFA<dfa.decisionNumber>(DFA):
  pass

  <@errorMethod()>

  <if(dfa.specialStateSTs)>
  def specialStateTransition(self_, s, input):
    # convince pylint that my self_ magic is ok ;
    # pylint: disable-msg=E0213

    # pretend we are a member of the recognizer
    # thus semantic predicates can be evaluated
    self = self_.recognizer

    _s = s

    <dfa.specialStateSTs:{state | if s == <i0>: <! compressed special state numbers 0..n-1 !>
    <state>}; separator="\n!>

  <if(backtracking)>
    if self._state.backtracking > 0:
      raise BacktrackingFailed

  <endif>
    nvae = NoViableAltException(self_.getDescription(), <dfa.decisionNumber>, _s, input)
    self_.error(nvae)
    raise nvae<\n>
  <endif>

```

>>

```
cyclicDFAInit(dfa) ::= <<
self.dfa<dfa.decisionNumber> = self.DFA<dfa.decisionNumber>(
  self, <dfa.decisionNumber>,
  eot = self.DFA<dfa.decisionNumber>_eot,
  eof = self.DFA<dfa.decisionNumber>_eof,
  min = self.DFA<dfa.decisionNumber>_min,
  max = self.DFA<dfa.decisionNumber>_max,
  accept = self.DFA<dfa.decisionNumber>_accept,
  special = self.DFA<dfa.decisionNumber>_special,
  transition = self.DFA<dfa.decisionNumber>_transition
)<\n>
>>
```

```
/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
LA<decisionNumber>_<stateNumber> = input.LA(1)<\n>
<if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
index<decisionNumber>_<stateNumber> = input.index()
input.rewind()<\n>
<endif>
s = -1
<edges; separator="\n">
<if(semPredState)> <! return input cursor to state before we rewound !>
input.seek(index<decisionNumber>_<stateNumber>)<\n>
<endif>
if s >= 0:
  return s
>>
```

```
/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if (<labelExpr>><if(predicates)> and (<predicates>)<endif>:
  s = <targetStateNumber><\n>
>>
```

```
/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
se:
  s = <targetStateNumber><\n>
```

>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "((<left>) and (<right>))"

orPredicates(operands) ::= "<operands; separator=\" or \">"

notPredicate(pred) ::= "not (<evalPredicate(pred, { })>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "self.<pred>()"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber> == <atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable

* somewhere. Must ask for the lookahead directly.

*/

isolatedLookaheadTest(atom,k,atomAsInt) ::= "self.input.LA(<k>) == <atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%

<lower> \<= LA<decisionNumber>_<stateNumber> \<= <upper>

%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "<lower> \<= self.input.LA(<k>) \<= <upper>)"

lookaheadSetTest(values,k,valuesAsInt) ::= <%

LA<decisionNumber>_<stateNumber> in {<values; separator=",">}

%>

isolatedLookaheadSetTest(values,k,valuesAsInt) ::= <%

self.input.LA(<k>) in {<values; separator=",">}

%>

lookaheadVarName(k) ::= "LA<decisionNumber>_<stateNumber>"

isolatedLookaheadVarName(k) ::= "self.input.LA(<k>)"

setTest(ranges) ::= "<ranges; separator=\" or \">"

// A T T R I B U T E S

globalAttributeScopeClass(scope) ::= <<

<if(scope)>

<if(scope.attributes)>

class <scope.name>_scope(object):

```

def __init__(self):
    <scope.attributes:{it | self.<it.decl> = None}; separator="\n">

<endif>
<endif>
>>

globalAttributeScopeStack(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
self.<scope.name>_stack = []<\n>
<endif>
<endif>
>>

ruleAttributeScopeClass(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
class <scope.name>_scope(object):
    def __init__(self):
        <scope.attributes:{it | self.<it.decl> = None}; separator="\n">

<endif>
<endif>
>>

ruleAttributeScopeStack(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
self.<scope.name>_stack = []<\n>
<endif>
<endif>
>>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<d.name><endif>
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <<
<label.label.text> = None
>>

returnStructName(r) ::= "<r.name>_return"

/** Define a return struct for a rule if the code needs to access its
* start/stop tokens, tree stuff, attributes, ... Leave a hole for
* subgroups to stick in members.

```

```

*/
returnScope(scope) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
class <ruleDescriptor:returnStructName()>(<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope):
  def __init__(self):
    super().__init__()

    <if(scope)><scope.attributes:{it | self.<it.decl> = None }; separator="\n"><endif>
    <@ruleReturnInit()>

    <@ruleReturnMembers()>

<endif>
>>

parameterScope(scope) ::= <<
<if(scope)><scope.attributes:{it | <it.decl> }; separator=", "><endif>
>>

parameterAttributeRef(attr) ::= "<attr.name>"
parameterSetAttributeRef(attr,expr) ::= "<attr.name> = <expr>"

scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
self.<scope>_stack[-<negIndex>].<attr.name>
<else>
<if(index)>
self.<scope>_stack[<index>].<attr.name>
<else>
self.<scope>_stack[-1].<attr.name>
<endif>
<endif>
%>

/* not applying patch because of bug in action parser!

<if(negIndex)>
((len(self.<scope>_stack) - <negIndex> - 1) >= 0 and [self.<scope>_stack[-<negIndex>].<attr.name>] or [None])[0]
<else>
<if(index)>
((<index> < len(self.<scope>_stack)) and [self.<scope>_stack[<index>].<attr.name>] or [None])[0]
<else>
((len(self.<scope>_stack) > 0) and [self.<scope>_stack[-1].<attr.name>] or [None])[0]
<endif>
<endif>

*/

```

```

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
<!FIXME: this seems not to be used by ActionTranslator...!>
self.<scope>_stack[-<negIndex>].<attr.name> = <expr>
<else>
<if(index)>
<!FIXME: this seems not to be used by ActionTranslator...!>
self.<scope>_stack[<index>].<attr.name> = <expr>
<else>
self.<scope>_stack[-1].<attr.name> = <expr>
<endif>
<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like { $function.size()>0 && $function::name.equals("foo") }?
 */
isolatedDynamicScopeRef(scope) ::= "self.<scope>_stack"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
((<scope> is not None) and [<scope>.<attr.name>] or [None])[0]
<else>
<scope>
<endif>
%>

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
%>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name> = <expr>
<else>
<attr.name> = <expr>
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

```

```

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach; and they are evaluated early;
// they cannot see TREE_PARSER or PARSER attributes for example. :(

tokenLabelPropertyRef_text(scope,attr) ::= "<scope>.text"
tokenLabelPropertyRef_type(scope,attr) ::= "<scope>.type"
tokenLabelPropertyRef_line(scope,attr) ::= "<scope>.line"
tokenLabelPropertyRef_pos(scope,attr) ::= "<scope>.charPositionInLine"
tokenLabelPropertyRef_channel(scope,attr) ::= "<scope>.channel"
tokenLabelPropertyRef_index(scope,attr) ::= "<scope>.index"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>.start"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>.stop"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>.tree"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
((<scope> is not None) and [self.input.getTokenStream().toString(
  self.input.getTreeAdaptor().getTokenStartIndex(<scope>.start),
  self.input.getTreeAdaptor().getTokenStopIndex(<scope>.start)
)] or [None])[0]
<else>
((<scope> is not None) and [self.input.toString(<scope>.start,<scope>.stop)] or [None])[0]
<endif>
%>
ruleLabelPropertyRef_st(scope,attr) ::= "((<scope> is not None) and [<scope>.st] or [None])[0]"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "((<scope> is not None) and [<scope>.type] or [0])[0]"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "((<scope> is not None) and [<scope>.line] or [0])[0]"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "((<scope> is not None) and [<scope>.charPositionInLine] or [0])[0]"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "((<scope> is not None) and [<scope>.channel] or [0])[0]"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "((<scope> is not None) and [<scope>.index] or [0])[0]"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "((<scope> is not None) and [<scope>.text] or [None])[0]"
lexerRuleLabelPropertyRef_int(scope,attr) ::= "((<scope> is not None) and [int(<scope>.text)] or [0])[0]"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.start"
rulePropertyRef_stop(scope,attr) ::= "retval.stop" //mmm... or input.LT(-1)??
rulePropertyRef_tree(scope,attr) ::= "retval.tree"
rulePropertyRef_text(scope,attr) ::= "self.input.toString(retval.start, self.input.LT(-1))"

```



```

rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "self.text"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "self._state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "self._state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "self._state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(self.getCharIndex()-1)"
lexerRulePropertyRef_int(scope,attr) ::= "int(<scope>.text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree =<expr>"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st =<expr>"

/** How to execute an action (only when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>
<if(actions.(actionScope).synpredgate)>
if <actions.(actionScope).synpredgate>:
    pass
    <action>

<else>
if <actions.(actionScope).synpredgate>:
    pass
    <action>

<endif>
<else>
#action start
<action>
#action end
<endif>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

codeFileExtension() ::= ".py"

true_value() ::= "True"

```

false_value() ::= "False"

Found in path(s):

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/Python3/Python3.stg

No license file was found, but licenses were detected in source scan.

/*

* [The "BSD license"]

* Copyright (c) 2011 Terence Parr

* All rights reserved.

*

* Conversion to C#:

* Copyright (c) 2011 Sam Harwell, Pixel Mine, Inc.

* All rights reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in the

* documentation and/or other materials provided with the distribution.

* 3. The name of the author may not be used to endorse or promote products

* derived from this software without specific prior written permission.

*

* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.

* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,

* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT

* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,

* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY

* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT

* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF

* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/** Templates for building ASTs during normal parsing.

*

* Deal with many combinations. Dimensions are:

* Auto build or rewrite

* no label, label, list label (label/no-label handled together)

* child, root

* token, set, rule, wildcard

*

* The situation is not too bad as rewrite (->) usage makes ^ and !

```

* invalid. There is no huge explosion of combinations.
*/

@rule.setErrorReturnValue() ::= <<
retval.Tree = (<ASTLabelType>)adaptor.ErrorNode(input, retval.Start, input.LT(-1), re);
<! System.out.WriteLine("<ruleName> returns "+((CommonTree)retval.tree).toStringTree()); !>
>>

// TOKEN AST STUFF

/** ID and output=AST */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <%
<super.tokenRef(...)>
<if(!ruleDescriptor.isSynPred)>
<if(backtracking)><\n>if (state.backtracking == 0) {<endif>
<\n><label>_tree = <createNodeFromToken(...)>;
<\n>adaptor.AddChild(root_0, <label>_tree);
<if(backtracking)><\n>}<endif>
<endif>
%>

/** ID! and output=AST (same as plain tokenRef) */
tokenRefBang(token,label,elementIndex,terminalOptions={}) ::= "<super.tokenRef(...)>"

/** ID^ and output=AST */
tokenRefRuleRoot(token,label,elementIndex,terminalOptions={}) ::= <%
<super.tokenRef(...)>
<if(!ruleDescriptor.isSynPred)>
<if(backtracking)><\n>if (<actions.(actionScope).synpredgate>) {<endif>
<\n><label>_tree = <createNodeFromToken(...)>;
<\n>root_0 = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_0);
<if(backtracking)><\n>}<endif>
<endif>
%>

/** ids+=ID! and output=AST */
tokenRefBangAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRefBang(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** label+=TOKEN when output=AST but not rewrite alt */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match label+=TOKEN^ when output=AST but not rewrite alt */

```

```

tokenRefRuleRootAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<tokenRefRuleRoot(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

// SET AST

// the match set stuff is interesting in that it uses an argument list
// to pass code to the default matchSet; another possible way to alter
// inherited code. I don't use the region stuff because I need to pass
// different chunks depending on the operator. I don't like making
// the template name have the operator as the number of templates gets
// large but this is the most flexible--this is as opposed to having
// the code generator call matchSet then add root code or ruleroot code
// plus list label plus ... The combinations might require complicated
// rather than just added on code. Investigate that refactoring when
// I have more time.

matchSet(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= <<
<super.matchSet(postmatchCode={ <if(!ruleDescriptor.isSynPred)><if(backtracking)>if
(<actions.(actionScope).synpredgate>) <endif>adaptor.AddChild(root_0, <createNodeFromToken(...)>;<endif>},
...)>
>>

matchRuleBlockSet(s,label,elementIndex,postmatchCode,treeLevel="0",terminalOptions={ }) ::= <<
<matchSet(...)>
>>

matchSetBang(s,label,elementIndex,postmatchCode,terminalOptions={ }) ::= "<super.matchSet(...)>"

// note there is no matchSetTrack because -> rewrites force sets to be
// plain old blocks of alts: (A|B|...|C)

matchSetRuleRoot(s,label,elementIndex,debug,terminalOptions={ }) ::= <<
<if(label)>
<label>=(<labelType>)input.LT(1);
<endif>
<super.matchSet(postmatchCode={ <if(!ruleDescriptor.isSynPred)><if(backtracking)>if
(<actions.(actionScope).synpredgate>) <endif>root_0 =
(<ASTLabelType>)adaptor.BecomeRoot(<createNodeFromToken(...)>, root_0);<endif>}, ...)>
>>

// RULE REF AST

/** rule when output=AST */
ruleRef(rule,label,elementIndex,args,scope) ::= <%
<super.ruleRef(...)>
<if(!ruleDescriptor.isSynPred)>

```

```

<\n><if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>adaptor.AddChild(root_0, <label>.Tree);
<endif>
%>

/** rule! is same as normal rule ref */
ruleRefBang(rule,label,elementIndex,args,scope) ::= "<super.ruleRef(...)>"

/** rule^ */
ruleRefRuleRoot(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) <endif>root_0 =
(<ASTLabelType>)adaptor.BecomeRoot(<label>.Tree, root_0);
>>

/** x+=rule when output=AST */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

/** x+=rule! when output=AST is a rule ref with list addition */
ruleRefBangAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefBang(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

/** x+=rule^ */
ruleRefRuleRootAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<listLabelElem(elem={<label>.Tree},elemType=ASTLabelType,...)>
>>

// WILDCARD AST

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>
<if(!ruleDescriptor.isSynPred)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.Create(<label>);
adaptor.AddChild(root_0, <label>_tree);
<if(backtracking)>}<endif>
<endif>
>>

wildcardBang(label,elementIndex) ::= "<super.wildcard(token=[],...)>"

wildcardRuleRoot(token,label,elementIndex,terminalOptions={ }) ::= <<
<super.wildcard(...)>

```

```

<if(!ruleDescriptor.isSynPred)>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
<label>_tree = (<ASTLabelType>)adaptor.Create(<label>);
root_0 = (<ASTLabelType>)adaptor.BecomeRoot(<label>_tree, root_0);
<if(backtracking)>}<endif>
<endif>
>>

createNodeFromToken(label,terminalOptions={ }) ::= <%
<if(terminalOptions.node)>
new
<terminalOptions.node><(if(terminalOptions.type))<terminalOptions.type>,<endif><label><(if(terminalOptions.text))>,<terminalOptions.text; format="string"><endif>
<else>
(<ASTLabelType>)adaptor.Create(<(if(terminalOptions.type))<terminalOptions.type>,<endif><label><(if(terminalOptions.text))>,<terminalOptions.text; format="string"><endif>)
<endif>
%>

ruleCleanUp() ::= <<
<super.ruleCleanUp()>
<if(backtracking)>if (<actions.(actionScope).synpredgate>) {<endif>
retval.Tree = (<ASTLabelType>)adaptor.RulePostProcessing(root_0);
adaptor.SetTokenBoundaries(retval.Tree, retval.Start, retval.Stop);
<if(backtracking)>}<endif>
>>

```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp3/ASTParser.stg
```

No license file was found, but licenses were detected in source scan.

```
/*
```

```
[The "BSD license"]
```

```
Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
```

```
http://www.temporal-wave.com
```

```
http://www.linkedin.com/in/jimidle
```

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

*/

/*

* This code generating template and the associated C runtime was produced by:

* Jim Idle jim|hereisanat|idle|dotgoeshere|ws.

* If it causes the destruction of the Universe, it will be pretty cool so long as

* I am in a different one at the time.

*/

cTypeInitMap ::= [

"int" : "0", // Integers start out being 0

"long" : "0", // Longs start out being 0

"float" : "0.0", // Floats start out being 0

"double" : "0.0", // Doubles start out being 0

"ANTLR3_BOOLEAN" : "ANTLR3_FALSE", // Booleans start out being Antlr C for false

"byte" : "0", // Bytes start out being 0

"short" : "0", // Shorts start out being 0

"char" : "0" // Chars start out being 0

]

leadIn(type) ::=

<<

/** \file

* This <type> file was generated by \$ANTLR version <ANTLRVersion>

*

* - From the grammar source file : <fileName>

* - On : <generatedTimestamp>

<if(LEXER)>

* - for the lexer : <name>Lexer

<endif>

<if(PARSER)>

* - for the parser : <name>Parser

<endif>

<if(TREE_PARSER)>

* - for the tree parser : <name>TreeParser

<endif>

```

*
* Editing it, at least manually, is not wise.
*
* C language generator and runtime by Jim Idle, jimi|hereisanat|idle|dotgoeshere|ws.
*
*
>>

/** The overall file structure of a recognizer; stores methods for rules
 * and cyclic DFAs plus support code.
 */
outputFile( LEXER,
            PARSER,
            TREE_PARSER,
            actionScope,
            actions,
            docComment,
            recognizer,
            name,
            tokens,
            tokenNames,
            rules,
            cyclicDFAs,
            bitsets,
            buildTemplate,
            buildAST,
            rewriteMode,
            profile,
            backtracking,
            synpreds,
            memoize,
            numRules,
            fileName,
            ANTLRVersion,
            generatedTimestamp,
            trace,
            scopes,
            superClass,
            literals
            ) ::=

<<
<leadIn("C source")>
*/
// [The "BSD license"]
// Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
// http://www.temporal-wave.com
// http://www.linkedin.com/in/jimidle
//

```



```

// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. The name of the author may not be used to endorse or promote products
// derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
// IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
// OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
// IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
// NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
// THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```

```
<if(actions.(actionScope).header)>
```

```

/* =====
* This is what the grammar programmer asked us to put at the top of every file.
*/

```

```
<actions.(actionScope).header>
```

```
/* End of Header action.
```

```

* =====
*/

```

```
<endif>
```

```

/* -----
* Include the ANTLR3 generated header file.
*/

```

```
#include "<name>.h"
```

```
<actions.(actionScope).postinclude>
```

```
/* ----- */
```

```
<docComment>
```

```
<if(literals)>
```

```
/** String literals used by <name> that we must do things like MATCHS() with.
```

```
* C will normally just lay down 8 bit characters, and you can use L"xxx" to
```

```
* get wchar_t, but wchar_t is 16 bits on Windows, which is not UTF32 and so
```

```

* we perform this little trick of defining the literals as arrays of UINT32
* and passing in the address of these.
*/
<literals:{it | static ANTLR3_UCHAR lit_<i>[] = <it>;}; separator="\n">

<endif>

/* MACROS that hide the C interface implementations from the
* generated code, which makes it a little more understandable to the human eye.
* I am very much against using C pre-processor macros for function calls and bits
* of code as you cannot see what is happening when single stepping in debuggers
* and so on. The exception (in my book at least) is for generated code, where you are
* not maintaining it, but may wish to read and understand it. If you single step it, you know that input()
* hides some indirect calls, but is always referring to the input stream. This is
* probably more readable than ctx->input->istream->input(snarfle0->blarg) and allows me to rejig
* the runtime interfaces without changing the generated code too often, without
* confusing the reader of the generated output, who may not wish to know the gory
* details of the interface inheritance.
*/

#define CTX ctx

/* Aids in accessing scopes for grammar programmers
*/
#undef SCOPE_TYPE
#undef SCOPE_STACK
#undef SCOPE_TOP
#define SCOPE_TYPE(scope) p<name>_##scope##_SCOPE
#define SCOPE_STACK(scope) p<name>_##scope##_Stack
#define SCOPE_TOP(scope) ctx->p<name>_##scope##_Top
#define SCOPE_SIZE(scope) ctx->p<name>_##scope##_Stack_limit
#define SCOPE_INSTANCE(scope, i) (ctx->SCOPE_STACK(scope)->get(ctx->SCOPE_STACK(scope),i))

<if(LEXER)>

/* Macros for accessing things in a lexer
*/
#undef LEXER
#undef RECOGNIZER
#undef RULEMEMO
#undef GETCHARINDEX
#undef GETLINE
#undef GETCHARPOSITIONINLINE
#undef EMIT
#undef EMITNEW

```

```

#undef MATCHC
#undef MATCHS
#undef MATCHRANGE
#undef LTOKEN
#undef HASFAILED
#undef FAILEDFLAG
#undef INPUT
#undef STRSTREAM
#undef LA
#undef HASEXCEPTION
#undef EXCEPTION
#undef CONSTRUCTEX
#undef CONSUME
#undef LRECOVER
#undef MARK
#undef REWIND
#undef REWINDLAST
#undef BACKTRACKING
#undef MATCHANY
#undef MEMOIZE
#undef HAVEPARSEDRULE
#undef GETTEXT
#undef INDEX
#undef SEEK
#undef PUSHSTREAM
#undef POPSTREAM
#undef SETTEXT
#undef SETTEXT8

#define LEXER ctx->pLexer
#define RECOGNIZER LEXER->rec
#define LEXSTATE RECOGNIZER->state
#define TOKSOURCE LEXSTATE->tokSource
#define GETCHARINDEX() LEXER->getCharIndex(LEXER)
#define GETLINE() LEXER->getLine(LEXER)
#define GETTEXT() LEXER->getText(LEXER)
#define GETCHARPOSITIONINLINE() LEXER->getCharPositionInLine(LEXER)
#define EMIT() LEXSTATE->type = _type; LEXER->emit(LEXER)
#define EMITNEW(t) LEXER->emitNew(LEXER, t)
#define MATCHC(c) LEXER->matchc(LEXER, c)
#define MATCHS(s) LEXER->matchs(LEXER, s)
#define MATCHRANGE(c1,c2) LEXER->matchRange(LEXER, c1, c2)
#define MATCHANY() LEXER->matchAny(LEXER)
#define LTOKEN LEXSTATE->token
#define HASFAILED() (LEXSTATE->failed == ANTLR3_TRUE)
#define BACKTRACKING LEXSTATE->backtracking
#define FAILEDFLAG LEXSTATE->failed
#define INPUT LEXER->input

```

```

#define STRSTREAM INPUT
#define ISTREAM INPUT->istream
#define INDEX() ISTREAM->index(ISTREAM)
#define SEEK(n) ISTREAM->seek(ISTREAM, n)
#define EOF_TOKEN &(LEXSTATE->tokSource->eofToken)
#define HASEXCEPTION() (LEXSTATE->error == ANTLR3_TRUE)
#define EXCEPTION LEXSTATE->exception
#define CONSTRUCTEX() RECOGNIZER->exConstruct(RECOGNIZER)
#define LRECOVER() LEXER->recover(LEXER)
#define MARK() ISTREAM->mark(ISTREAM)
#define REWIND(m) ISTREAM->rewind(ISTREAM, m)
#define REWINDLAST() ISTREAM->rewindLast(ISTREAM)
#define MEMOIZE(ri,si) RECOGNIZER->memoize(RECOGNIZER, ri, si)
#define HAVEPARSEDRULE(r) RECOGNIZER->alreadyParsedRule(RECOGNIZER, r)
#define PUSHSTREAM(str) LEXER->pushCharStream(LEXER, str)
#define POPSTREAM() LEXER->popCharStream(LEXER)
#define SETTEXT(str) LEXSTATE->text = str
#define SKIP() LEXSTATE->token = &(TOKSOURCE->skipToken)
#define USER1 LEXSTATE->user1
#define USER2 LEXSTATE->user2
#define USER3 LEXSTATE->user3
#define CUSTOM LEXSTATE->custom
#define RULEMEMO LEXSTATE->ruleMemo
#define DBG RECOGNIZER->debugger

/* If we have been told we can rely on the standard 8 bit or UTF16 input
 * stream, then we can define our macros to use the direct pointers
 * in the input object, which is much faster than indirect calls. This
 * is really only significant to lexers with a lot of fragment rules (which
 * do not place LA(1) in a temporary at the moment) and even then
 * only if there is a lot of input (order of say 1M or so).
 */
#if defined(ANTLR3_INLINE_INPUT_8BIT) || defined(ANTLR3_INLINE_INPUT_UTF16)

# ifdef ANTLR3_INLINE_INPUT_8BIT

/* 8 bit character set */

# define NEXTCHAR ((pANTLR3_UINT8)(INPUT->nextChar))
# define DATAP ((pANTLR3_UINT8)(INPUT->data))

# else

# define NEXTCHAR ((pANTLR3_UINT16)(INPUT->nextChar))
# define DATAP ((pANTLR3_UINT16)(INPUT->data))

# endif

```

```

#define LA(n) ((NEXTCHAR + n) > (DATAP + INPUT->sizeBuf) ? ANTLR3_CHARSTREAM_EOF :
(ANTLR3_UCHAR)(*NEXTCHAR + n - 1))
#define CONSUME() //
{ //
if (NEXTCHAR < (DATAP + INPUT->sizeBuf)) //
{ //
INPUT->charPositionInLine++; //
if ((ANTLR3_UCHAR)(*NEXTCHAR) == INPUT->newlineChar) //
{ //
INPUT->line++; //
INPUT->charPositionInLine = 0; //
INPUT->currentLine = (void *) (NEXTCHAR + 1); //
} //
INPUT->nextChar = (void *) (NEXTCHAR + 1); //
} //
}

#else

// Pick up the input character by calling the input stream implementation.
//
#define CONSUME() INPUT->istream->consume(INPUT->istream)
#define LA(n) INPUT->istream->_LA(INPUT->istream, n)

#endif
<endif>

<if(PARSER)>
/* Macros for accessing things in the parser
*/

#undef PARSER
#undef RECOGNIZER
#undef HAVEPARSEDRULE
#undef MEMOIZE
#undef INPUT
#undef STRSTREAM
#undef HASEXCEPTION
#undef EXCEPTION
#undef MATCHT
#undef MATCHANYT
#undef FOLLOWSTACK
#undef FOLLOWPUSH
#undef FOLLOWPOP
#undef PRECOVER
#undef PREPORTERROR
#undef LA
#undef LT

```

```

#undef CONSTRUCTEX
#undef CONSUME
#undef MARK
#undef REWIND
#undef REWINDLAST
#undef PERRORRECOVERY
#undef HASFAILED
#undef FAILEDFLAG
#undef RECOVERFROMMISMATCHEDSET
#undef RECOVERFROMMISMATCHEDELEMENT
#undef INDEX
#undef ADAPTOR
#undef SEEK
#undef RULEMEMO
#undef DBG

#define PARSER ctx->pParser
#define RECOGNIZER PARSER->rec
#define PSRSTATE RECOGNIZER->state
#define HAVEPARSEDRULE(r) RECOGNIZER->alreadyParsedRule(RECOGNIZER, r)
#define MEMOIZE(ri,si) RECOGNIZER->memoize(RECOGNIZER, ri, si)
#define INPUT PARSER->tstream
#define STRSTREAM INPUT
#define ISTREAM INPUT->istream
#define INDEX() ISTREAM->index(INPUT->istream)
#define HASEXCEPTION() (PSRSTATE->error == ANTLR3_TRUE)
#define EXCEPTION PSRSTATE->exception
#define MATCHT(t, fs) RECOGNIZER->match(RECOGNIZER, t, fs)
#define MATCHANYT() RECOGNIZER->matchAny(RECOGNIZER)
#define FOLLOWSTACK PSRSTATE->following
#ifdef SKIP_FOLLOW_SETS
#define FOLLOWPUSH(x)
#define FOLLOWPOP()
#else
#define FOLLOWPUSH(x) FOLLOWSTACK->push(FOLLOWSTACK, ((void *)&(x))), NULL)
#define FOLLOWPOP() FOLLOWSTACK->pop(FOLLOWSTACK)
#endif
#define PRECOVER() RECOGNIZER->recover(RECOGNIZER)
#define PREPORTERROR() RECOGNIZER->reportError(RECOGNIZER)
#define LA(n) INPUT->istream->_LA(ISTREAM, n)
#define LT(n) INPUT->_LT(INPUT, n)
#define CONSTRUCTEX() RECOGNIZER->exConstruct(RECOGNIZER)
#define CONSUME() ISTREAM->consume(ISTREAM)
#define MARK() ISTREAM->mark(ISTREAM)
#define REWIND(m) ISTREAM->rewind(ISTREAM, m)
#define REWINDLAST() ISTREAM->rewindLast(ISTREAM)
#define SEEK(n) ISTREAM->seek(ISTREAM, n)
#define PERRORRECOVERY PSRSTATE->errorRecovery

```

```

#define FAILEDFLAG PSRSTATE->failed
#define HASFAILED() (FAILEDFLAG == ANTLR3_TRUE)
#define BACKTRACKING PSRSTATE->backtracking
#define RECOVERFROMMISMATCHEDSET(s) RECOGNIZER->recoverFromMismatchedSet(RECOGNIZER,
s)
#define RECOVERFROMMISMATCHEDELEMENT(e) RECOGNIZER-
>recoverFromMismatchedElement(RECOGNIZER, s)
#define ADAPTOR          ctx->adaptor
#define RULEMEMO         PSRSTATE->ruleMemo
#define DBG             RECOGNIZER->debugger

```

```
<endif>
```

```
<if(TREE_PARSER)>
```

```
/* Macros for accessing things in the parser
```

```
*/
```

```

#undef  PARSER
#undef  RECOGNIZER
#undef  HAVEPARSEDRULE
#undef  INPUT
#undef  STRSTREAM
#undef  HASEXCEPTION
#undef  EXCEPTION
#undef  MATCHT
#undef  MATCHANYT
#undef  FOLLOWSTACK
#undef  FOLLOWPUSH
#undef  FOLLOWPOP
#undef  PRECOVER
#undef  PREPORTERROR
#undef  LA
#undef  LT
#undef  CONSTRUCTEX
#undef  CONSUME
#undef  MARK
#undef  REWIND
#undef  REWINDLAST
#undef  PERRORRECOVERY
#undef  HASFAILED
#undef  FAILEDFLAG
#undef  RECOVERFROMMISMATCHEDSET
#undef  RECOVERFROMMISMATCHEDELEMENT
#undef  BACKTRACKING
#undef  ADAPTOR
#undef  RULEMEMO
#undef  SEEK
#undef  INDEX

```

```

#undef DBG

#define  PARSER      ctx->pTreeParser
#define  RECOGNIZER  PARSER->rec
#define  PSRSTATE    RECOGNIZER->state
#define  HAVEPARSEDRULE(r)  RECOGNIZER->alreadyParsedRule(RECOGNIZER, r)
#define  INPUT      PARSER->ctnstream
#define  ISTREAM    INPUT->tnstream->istream
#define  STRSTREAM  INPUT->tnstream
#define  HASEXCEPTION()  (PSRSTATE->error == ANTLR3_TRUE)
#define  EXCEPTION   PSRSTATE->exception
#define  MATCHT(t, fs)  RECOGNIZER->match(RECOGNIZER, t, fs)
#define  MATCHANYT()  RECOGNIZER->matchAny(RECOGNIZER)
#define  FOLLOWSTACK  PSRSTATE->following
#define  FOLLOWPUSH(x)  FOLLOWSTACK->push(FOLLOWSTACK, ((void *)&(x)), NULL)
#define  FOLLOWPOP()  FOLLOWSTACK->pop(FOLLOWSTACK)
#define  PRECOVER()  RECOGNIZER->recover(RECOGNIZER)
#define  PREPORTERROR()  RECOGNIZER->reportError(RECOGNIZER)
#define  LA(n)      ISTREAM->_LA(ISTREAM, n)
#define  LT(n)      INPUT->tnstream->_LT(INPUT->tnstream, n)
#define  CONSTRUCTEX()  RECOGNIZER->exConstruct(RECOGNIZER)
#define  CONSUME()  ISTREAM->consume(ISTREAM)
#define  MARK()     ISTREAM->mark(ISTREAM)
#define  REWIND(m)  ISTREAM->rewind(ISTREAM, m)
#define  REWINDLAST()  ISTREAM->rewindLast(ISTREAM)
#define  PERRORRECOVERY  PSRSTATE->errorRecovery
#define  FAILEDFLAG  PSRSTATE->failed
#define  HASFAILED()  (FAILEDFLAG == ANTLR3_TRUE)
#define  BACKTRACKING  PSRSTATE->backtracking
#define  RECOVERFROMMISMATCHEDSET(s)  RECOGNIZER->recoverFromMismatchedSet(RECOGNIZER,
s)
#define  RECOVERFROMMISMATCHEDELEMENT(e)  RECOGNIZER-
>recoverFromMismatchedElement(RECOGNIZER, s)
#define  ADAPTOR          INPUT->adaptor
#define  RULEMEMO        PSRSTATE->ruleMemo
#define  SEEK(n)         ISTREAM->seek(ISTREAM, n)
#define  INDEX()        ISTREAM->index(ISTREAM)
#define  DBG            RECOGNIZER->debugger

<endif>

#define  TOKTEXT(tok, txt)  tok, (pANTLR3_UINT8)txt

/* The 4 tokens defined below may well clash with your own #defines or token types. If so
* then for the present you must use different names for your defines as these are hard coded
* in the code generator. It would be better not to use such names internally, and maybe
* we can change this in a forthcoming release. I deliberately do not #undef these

```



```

* here as this will at least give you a redefined error somewhere if they clash.
*/
#define UP ANTLR3_TOKEN_UP
#define DOWN ANTLR3_TOKEN_DOWN
#define EOR ANTLR3_TOKEN_EOR
#define INVALID ANTLR3_TOKEN_INVALID

/* =====
* Functions to create and destroy scopes. First come the rule scopes, followed
* by the global declared scopes.
*/

<rules: {r |<if(r.ruleDescriptor.ruleScope)>
<ruleAttributeScopeFuncDecl(scope=r.ruleDescriptor.ruleScope)>
<ruleAttributeScopeFuncs(scope=r.ruleDescriptor.ruleScope)>
<endif>}>

<recognizer.scopes:{it | <if(it.isDynamicGlobalScope)>
<globalAttributeScopeFuncDecl(it)>
<globalAttributeScopeFuncs(it)>
<endif>}>

/* ===== */

/* =====
* Start of recognizer
*/

<recognizer>

/* End of code
* =====
*/

>>
headerFileExtension() ::= ".h"

headerFile( LEXER,
            PARSER,
            TREE_PARSER,
            actionScope,
            actions,
            docComment,
            recognizer,
            name,
            tokens,
            tokenNames,

```

```

    rules,
    cyclicDFAs,
    bitsets,
    buildTemplate,
    buildAST,
    rewriteMode,
    profile,
    backtracking,
    synpreds,
    memoize,
    numRules,
    fileName,
    ANTLRVersion,
    generatedTimestamp,
    trace,
    scopes,
    superClass,
    literals
) ::=
<<
<leadIn("C header")>
<if(PARSER)>
* The parser <mainName()>
<endif>
<if(LEXER)>
* The lexer <mainName()>
<endif>
<if(TREE_PARSER)>
* The tree parser <mainName()>
<endif>
has the callable functions (rules) shown below,
* which will invoke the code for the associated rule in the source grammar
* assuming that the input stream is pointing to a token/text stream that could begin
* this rule.
*
* For instance if you call the first (topmost) rule in a parser grammar, you will
* get the results of a full parse, but calling a rule half way through the grammar will
* allow you to pass part of a full token stream to the parser, such as for syntax checking
* in editors and so on.
*
* The parser entry points are called indirectly (by function pointer to function) via
* a parser context typedef p<name>, which is returned from a call to <name>New().
*
<if(LEXER)>
* As this is a generated lexer, it is unlikely you will call it 'manually'. However
* the methods are provided anyway.
*
<endif>

```

```

* The methods in p<name> are as follows:
*
* <rules:{r | <if(!r.ruleDescriptor.isSynPred)> - <headerReturnType(ruleDescriptor=r.ruleDescriptor,...)>
p<name>-><r.ruleDescriptor.name>(p<name><endif>}; separator="\n * ">
*
* The return type for any particular rule is of course determined by the source
* grammar file.
*/
// [The "BSD license"]
// Copyright (c) 2005-2009 Jim Idle, Temporal Wave LLC
// http://www.temporal-wave.com
// http://www.linkedin.com/in/jimidle
//
// All rights reserved.
//
// Redistribution and use in source and binary forms, with or without
// modification, are permitted provided that the following conditions
// are met:
// 1. Redistributions of source code must retain the above copyright
// notice, this list of conditions and the following disclaimer.
// 2. Redistributions in binary form must reproduce the above copyright
// notice, this list of conditions and the following disclaimer in the
// documentation and/or other materials provided with the distribution.
// 3. The name of the author may not be used to endorse or promote products
// derived from this software without specific prior written permission.
//
// THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
// IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
// OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
// IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
// INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
// NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
// DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
// THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
// (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
// THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

#ifndef _<name>_H
#define _<name>_H
<actions.(actionScope).preincludes>
/* =====
* Standard antlr3 C runtime definitions
*/
#include \<antlr3.h>

/* End of standard antlr 3 runtime definitions
* =====
*/

```

```

<actions.(actionScope).includes>

#ifdef __cplusplus
extern "C" {
#endif

// Forward declare the context typedef so that we can use it before it is
// properly defined. Delegators and delegates (from import statements) are
// interdependent and their context structures contain pointers to each other
// C only allows such things to be declared if you pre-declare the typedef.
//
typedef struct <name>_Ctx_struct <name>, * p<name>;

<if(recognizer.grammar.delegates)>
// Include delegate definition header files
//
<recognizer.grammar.delegates: {g|#include \<<g.recognizerName>.h>}; separator="\n">

<endif>

<actions.(actionScope).header>

#ifdef ANTLR3_WINDOWS
// Disable: Unreferenced parameter, - Rules with parameters that are not used
// constant conditional, - ANTLR realizes that a prediction is always true (synpred usually)
// initialized but unused variable - tree rewrite variables declared but not needed
// Unreferenced local variable - lexer rule declares but does not always use _type
// potentially uninitialized variable used - retval always returned from a rule
// unreferenced local function has been removed - susually getTokenNames or freeScope, they can go without
warnigns
//
// These are only really displayed at warning level /W4 but that is the code ideal I am aiming at
// and the codegen must generate some of these warnings by necessity, apart from 4100, which is
// usually generated when a parser rule is given a parameter that it does not use. Mostly though
// this is a matter of orthogonality hence I disable that one.
//
#pragma warning( disable : 4100 )
#pragma warning( disable : 4101 )
#pragma warning( disable : 4127 )
#pragma warning( disable : 4189 )
#pragma warning( disable : 4505 )
#pragma warning( disable : 4701 )
#endif
<if(backtracking)>

/* =====
* BACKTRACKING IS ENABLED

```

```

* =====
*/
<endif>

<rules:{r |<headerReturnScope(ruleDescriptor=r.ruleDescriptor,...)>>

<scopes:{it | <if(it.isDynamicGlobalScope)><globalAttributeScopeDecl(it)><endif>>
<rules:{r |<ruleAttributeScopeDecl(scope=r.ruleDescriptor.ruleScope)>>
<if(recognizer.grammar.delegators)>
// Include delegator definition header files
//
<recognizer.grammar.delegators: {g#include \<<g.recognizerName>.h>; separator="\n">

<endif>

/** Context tracking structure for <mainName()>
*/
struct <name>_Ctx_struct
{
    /** Built in ANTLR3 context tracker contains all the generic elements
    * required for context tracking.
    */
<if(PARSER)>
    pANTLR3_PARSER  pParser;
<endif>
<if(LEXER)>
    pANTLR3_LEXER  pLexer;
<endif>
<if(TREE_PARSER)>
    pANTLR3_TREE_PARSER  pTreeParser;
<endif>

<if(recognizer.grammar.delegates)>
<recognizer.grammar.delegates:
    {g|p<g.recognizerName> <g:delegateName(>;}; separator="\n">
<endif>
<if(recognizer.grammar.delegators)>
<recognizer.grammar.delegators:
    {g|p<g.recognizerName> <g:delegateName(>;}; separator="\n">
<endif>
<scopes:{it | <if(it.isDynamicGlobalScope)>
    <globalAttributeScopeDef(it)>
<endif>}; separator="\n\n">
<rules: {r |<if(r.ruleDescriptor.ruleScope)>
    <ruleAttributeScopeDef(scope=r.ruleDescriptor.ruleScope)>
<endif>}>

<if(LEXER)>

```

```

    <rules:{r | <if(!r.ruleDescriptor.isSynPred)><headerReturnType(ruleDescriptor=r.ruleDescriptor)>
(*m<r.ruleDescriptor.name>) (struct <name>_Ctx_struct * ctx<if(r.ruleDescriptor.parameterScope)>,
<endif><r.ruleDescriptor.parameterScope:parameterScope(>>);<endif>}; separator="\n">
<endif>
<if(!LEXER)>
    <rules:{r | <headerReturnType(ruleDescriptor=r.ruleDescriptor)> (*<r.ruleDescriptor.name>) (struct
<name>_Ctx_struct * ctx<if(r.ruleDescriptor.parameterScope)>,
<endif><r.ruleDescriptor.parameterScope:parameterScope(>>);}; separator="\n">
<! generate rule/method definitions for imported rules so they
appear to be defined in this recognizer. !>
    // Delegated rules
<recognizer.grammar.delegatedRules:{ruleDescriptor|
    <headerReturnType(ruleDescriptor)> (*<ruleDescriptor.name>)(struct <name>_Ctx_struct *
ctx<if(ruleDescriptor.parameterScope)>, <endif><r.ruleDescriptor.parameterScope:parameterScope(>>);};
separator="\n">
<endif>

    const char * (*getGrammarFileName());
    void (*reset) (struct <name>_Ctx_struct * ctx);
    void (*free) (struct <name>_Ctx_struct * ctx);
    <@members>
    <@end>
    <actions.(actionScope).context>
};

// Function prototypes for the constructor functions that external translation units
// such as delegators and delegates may wish to call.
//
ANTLR3_API p<name> <name>New (<inputType(>) instream<recognizer.grammar.delegators:{g|,
p<g.recognizerName> <g.delegateName(>>}&>);
ANTLR3_API p<name> <name>NewSSD (<inputType(>) instream,
pANTLR3_RECOGNIZER_SHARED_STATE state<recognizer.grammar.delegators:{g|, p<g.recognizerName>
<g.delegateName(>>}&>);
<if(!recognizer.grammar.grammarIsRoot)>
extern pANTLR3_UINT8 <recognizer.grammar.composite.rootGrammar.recognizerName>TokenNames[];
<endif>

/** Symbolic definitions of all the tokens that the <grammarType(>) will work with.
* \{
*
* Antlr will define EOF, but we can't use that as it is too common in
* in C header files and that would be confusing. There is no way to filter this out at the moment
* so we just undef it here for now. That isn't the value we get back from C recognizers
* anyway. We are looking for ANTLR3_TOKEN_EOF.
*/
#ifdef EOF
#undef EOF

```

```

#endif
#ifdef Tokens
#undef Tokens
#endif
<tokens:{it | #define <it.name>    <it.type>}; separator="\n">
#ifdef EOF
#undef EOF
#define EOF ANTLR3_TOKEN_EOF
#endif

#ifndef TOKENSOURCE
#define TOKENSOURCE(lxr) lxr->pLexer->rec->state->tokSource
#endif

/* End of token definitions for <name>
* =====
*/
/** \} */

#ifdef __cplusplus
}
#endif

#endif

/* END - Note:Keep extra line feed to satisfy UNIX systems */

>>

inputType() ::= <<
<if(LEXER)>
pANTLR3_INPUT_STREAM
<endif>
<if(PARSER)>
pANTLR3_COMMON_TOKEN_STREAM
<endif>
<if(TREE_PARSER)>
pANTLR3_COMMON_TREE_NODE_STREAM
<endif>
>>

grammarType() ::= <<
<if(PARSER)>
parser
<endif>
<if(LEXER)>
lexer
<endif>

```

```
<if(TREE_PARSER)>
tree parser
<endif>
>>
```

```
mainName() ::= <<
<if(PARSER)>
<name>
<endif>
<if(LEXER)>
<name>
<endif>
<if(TREE_PARSER)>
<name>
<endif>
>>
```

```
headerReturnScope(ruleDescriptor) ::= "<returnScope(...)>"
```

```
headerReturnType(ruleDescriptor) ::= <<
<if(LEXER)>
<if(!ruleDescriptor.isSynPred)>
void
<else>
<returnType()>
<endif>
<else>
<returnType()>
<endif>
>>
```

```
// Produce the lexer output
//
```

```
lexer( grammar,
name,
tokens,
scopes,
rules,
numRules,
filterMode,
superClass,
labelType="pANTLR3_COMMON_TOKEN") ::= <<
```

```
<if(filterMode)>
```

```
/* Forward declare implementation function for ANTLR3_TOKEN_SOURCE interface when
```

```
* this is a filter mode lexer.
```

```
*/
```

```
static pANTLR3_COMMON_TOKEN <name>NextToken (pANTLR3_TOKEN_SOURCE toksource);
```



```

/* Override the normal MEMOIZE and HAVEALREADYPARSED macros as this is a filtering
* lexer. In filter mode, the memoizing and backtracking are gated at BACKTRACKING > 1 rather
* than just BACKTRACKING. In some cases this might generate code akin to:
* if (BACKTRACKING) if (BACKTRACKING > 1) memoize.
* However, I assume that the C compilers/optimizers are smart enough to work this one out
* these days - Jim
*/
#undef MEMOIZE
#define MEMOIZE(ri,si) if (BACKTRACKING>1) { RECOGNIZER->memoize(RECOGNIZER, ri, si) }
#undef HAVEPARSEDRULE
#define HAVEPARSEDRULE(r) if (BACKTRACKING>1) { RECOGNIZER->alreadyParsedRule(RECOGNIZER,
r) }
<endif>

/* Forward declare the locally static matching functions we have generated and any predicate functions.
*/
<rules:{r | static ANTLR3_INLINE <headerReturnType(ruleDescriptor=r.ruleDescriptor)>
<if(!r.ruleDescriptor.isSynPred)>m<endif><r.ruleDescriptor.name> (p<name>
ctx<if(r.ruleDescriptor.parameterScope)>, <endif><r.ruleDescriptor.parameterScope:parameterScope(>>);};
separator="\n">
static void <name>Free(p<name> ctx);

/* =====
* Lexer matching rules end.
* =====
*/

<scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScope(it)><endif>}>

<actions.lexer.members>

static void
<name>Free (p<name> ctx)
{
<if(memoize)>
if (RULEMEMO != NULL)
{
RULEMEMO->free(RULEMEMO);
RULEMEMO = NULL;
}
<endif>
<if(grammar.directDelegates)>
// Free the lexers that we delegated to
// functions to. NULL the state so we only free it once.
//
<grammar.directDelegates:
{g|ctx-><g:delegateName(>->pLexer->rec->state = NULL;

```

```

    ctx-><g:delegateName(>->free(ctx-><g:delegateName(>));}; separator="\n">
<endif>
    LEXER->free(LEXER);

    ANTLR3_FREE(ctx);
}

static void
<name>Reset (p<name> ctx)
{
    RECOGNIZER->reset(RECOGNIZER);
}

/** \brief Name of the grammar file that generated this code
*/
static const char fileName[] = "<fileName>";

/** \brief Return the name of the grammar file that generated this code.
*/
static const char * getGrammarFileName()
{
    return fileName;
}

<if(filterMode)>
    <filteringNextToken(>
<endif>

/** \brief Create a new lexer called <name>
*
* \param[in] instream Pointer to an initialized input stream
* \return
* - Success p<name> initialized for the lex start
* - Fail NULL
*/
ANTLR3_API p<name> <name>New
(<inputType(> instream<grammar.delegators:{g|, p<g.recognizerName> <g:delegateName(>}>>)
{
    // See if we can create a new lexer with the standard constructor
    //
    return <name>NewSSD(instream, NULL<grammar.delegators:{g|, <g:delegateName(>}>>);
}

/** \brief Create a new lexer called <name>
*
* \param[in] instream Pointer to an initialized input stream
* \param[state] state Previously created shared recognizer stat
* \return

```

```

* - Success p<name> initialized for the lex start
* - Fail NULL
*/
ANTLR3_API p<name> <name>NewSSD
(pANTLR3_INPUT_STREAM instream, pANTLR3_RECOGNIZER_SHARED_STATE
state<grammar.delegators:{g|, p<g.recognizerName> <g.delegateName()>>})
{
    p<name> ctx; // Context structure we will build and return

    ctx = (p<name>) ANTLR3_CALLOC(1, sizeof(<name>));

    if (ctx == NULL)
    {
        // Failed to allocate memory for lexer context
        return NULL;
    }

    /* -----
    * Memory for basic structure is allocated, now to fill in
    * in base ANTLR3 structures. We initialize the function pointers
    * for the standard ANTLR3 lexer function set, but upon return
    * from here, the programmer may set the pointers to provide custom
    * implementations of each function.
    *
    * We don't use the macros defined in <name>.h here so you can get a sense
    * of what goes where.
    */

    /* Create a base lexer, using the supplied input stream
    */
    ctx->pLexer = antlr3LexerNewStream(ANTLR3_SIZE_HINT, instream, state);

    /* Check that we allocated the memory correctly
    */
    if (ctx->pLexer == NULL)
    {
        ANTLR3_FREE(ctx);
        return NULL;
    }
<if(memoize)>
<if(grammar.grammarIsRoot)>
    // Create a LIST for recording rule memos.
    //
    ctx->pLexer->rec->ruleMemo = antlr3IntTrieNew(15); /* 16 bit depth is enough for 32768 rules! */
<endif>
<endif>

    /* Install the implementation of our <name> interface

```

```

*/
<rules:{r | <if(!r.ruleDescriptor.isSynPred)>ctx->m<r.ruleDescriptor.name> =
m<r.ruleDescriptor.name>;<endif>}>; separator="\n">

/** When the nextToken() call is made to this lexer's pANTLR3_TOKEN_SOURCE
 * it will call mTokens() in this generated code, and will pass it the ctx
 * pointer of this lexer, not the context of the base lexer, so store that now.
 */
ctx->pLexer->ctx = ctx;

/**Install the token matching function
 */
ctx->pLexer->mTokens = (void (*)(void *))(mTokens);

ctx->getGrammarFileName = getGrammarFileName;
ctx->free = <name>Free;
ctx->reset = <name>Reset;

<if(grammar.directDelegates)>
// Initialize the lexers that we are going to delegate some
// functions to.
//
<grammar.directDelegates:
    {g|ctx-><g:delegateName()> = <g.recognizerName>NewSSD(instream, ctx->pLexer->rec->state,
ctx->grammar.delegates:{g, <g:delegateName()>}>);}; separator="\n">
<endif>
<if(grammar.delegates)>
// Install the pointers back to lexers that will delegate us to perform certain functions
// for them.
//
<grammar.delegates:
    {g|ctx-><g:delegateName()> = <g:delegateName()>;}; separator="\n">
<endif>
<if(filterMode)>
/* We have filter mode turned on, so install the filtering nextToken function
 */
ctx->pLexer->rec->state->tokSource->nextToken = <name>NextToken;
<endif>
<actions.lexer.apifuncs>

/* Return the newly built lexer to the caller
 */
return ctx;
}
<if(cyclicDFAs)>

/* =====
 * DFA tables for the lexer

```

```

*/
<cyclicDFAs:cyclicDFA(>< ! dump tables for all DFA !>
/* =====
* End of DFA tables for the lexer
*/
<endif>

/* =====
* Functions to match the lexer grammar defined tokens from the input stream
*/

<rules; separator="\n\n">

/* =====
* Lexer matching rules end.
* =====
*/
<if(synpreds)>

/* =====
* Lexer syntactic predicates
*/
<synpreds: {p | <lexerSynpred(predname=p)>}>
/* =====
* Lexer syntactic predicates end.
* =====
*/
<endif>

/* End of Lexer code
* =====
* =====
*/

>>

```

```

filteringNextToken() ::= <<
/** An override of the lexer's nextToken() method that backtracks over mTokens() looking
* for matches in lexer filterMode. No error can be generated upon error; just rewind, consume
* a token and then try again. BACKTRACKING needs to be set as well.
* Make rule memoization happen only at levels above 1 as we start mTokens
* at BACKTRACKING==1.
*/
static pANTLR3_COMMON_TOKEN
<name>NextToken(pANTLR3_TOKEN_SOURCE toksource)
{
    pANTLR3_LEXER lexer;

```

```

pANTLR3_RECOGNIZER_SHARED_STATE state;

lexer = (pANTLR3_LEXER)(toksource->super);
state = lexer->rec->state;

/* Get rid of any previous token (token factory takes care of
 * any deallocation when this token is finally used up.
 */
state ->token = NULL;
state ->error = ANTLR3_FALSE; /* Start out without an exception */
state ->failed = ANTLR3_FALSE;

/* Record the start of the token in our input stream.
 */
state->tokenStartCharIndex = lexer->input->istream->index(lexer->input->istream);
state->tokenStartCharPositionInLine = lexer->input->getCharPositionInLine(lexer->input);
state->tokenStartLine = lexer->input->getLine(lexer->input);
state->text = NULL;

/* Now call the matching rules and see if we can generate a new token
 */
for (;;)
{
if (lexer->input->istream->_LA(lexer->input->istream, 1) == ANTLR3_CHARSTREAM_EOF)
{
/* Reached the end of the stream, nothing more to do.
 */
pANTLR3_COMMON_TOKEN teof = &(toksource->eofToken);

teof->setStartIndex (teof, lexer->getCharIndex(lexer));
teof->setStopIndex (teof, lexer->getCharIndex(lexer));
teof->setLine (teof, lexer->getLine(lexer));
return teof;
}

state->token = NULL;
state->error = ANTLR3_FALSE; /* Start out without an exception */

{
ANTLR3_MARKER m;

m = lexer->input->istream->mark(lexer->input->istream);
state->backtracking = 1; /* No exceptions */
state->failed = ANTLR3_FALSE;

/* Call the generated lexer, see if it can get a new token together.
 */
lexer->mTokens(lexer->ctx);

```

```

state->backtracking = 0;

<! mTokens backtracks with synpred at BACKTRACKING==2
and we set the synpredgate to allow actions at level 1. !>

if (state->failed == ANTLR3_TRUE)
{
lexer->input->istream->rewind(lexer->input->istream, m);
lexer->input->istream->consume(lexer->input->istream); <! advance one char and try again !>
}
else
{
lexer->emit(lexer); /* Assemble the token and emit it to the stream */
return state->token;
}
}
}
}
}
>>

```

```

actionGate() ::= "BACKTRACKING==0"

```

```

filteringActionGate() ::= "BACKTRACKING==1"

```

```

/** How to generate a parser */
genericParser( grammar,
name,
scopes,
tokens,
tokenNames,
rules,
numRules,
bitsets,
inputStreamType,
superClass,
labelType,
members,
rewriteElementType, filterMode,
ASTLabelType="pANTLR3_BASE_TREE"
) ::= <<

```

```

<if(grammar.grammarIsRoot)>
/** \brief Table of all token names in symbolic order, mainly used for
* error reporting.
*/
pANTLR3_UINT8 <name>TokenNames[<length(tokenNames)>+4]
= {

```

```

    (pANTLR3_UINT8) "\<invalid>", /* String to print to indicate an invalid token */
    (pANTLR3_UINT8) "\<EOR>",
    (pANTLR3_UINT8) "\<DOWN>",
    (pANTLR3_UINT8) "\<UP>",
    <tokenNames:{it |(pANTLR3_UINT8) <it>}; separator=",\n">
};
<endif>

<@members>

<@end>
<rules:{r |<ruleAttributeScopeFuncMacro(scope=r.ruleDescriptor.ruleScope)>>}>
<scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScopeFuncMacro(it)><endif>}>

// Forward declare the locally static matching functions we have generated.
//
<rules:{r | static <headerReturnType(ruleDescriptor=r.ruleDescriptor)> <r.ruleDescriptor.name> (p<name>
ctx<if(r.ruleDescriptor.parameterScope)>, <endif><r.ruleDescriptor.parameterScope:parameterScope()>);
separator="\n">
static void <name>Free(p<name> ctx);
static void <name>Reset (p<name> ctx);

<if(!LEXER)>
<! generate rule/method definitions for imported rules so they
appear to be defined in this recognizer. !>
<if(recognizer.grammar.delegatedRules)>
// Delegated rules
//
<recognizer.grammar.delegatedRules:{ruleDescriptor|static <headerReturnType(ruleDescriptor)>
<ruleDescriptor.name>(p<name> ctx<if(ruleDescriptor.parameterScope)>,
<endif><r.ruleDescriptor.parameterScope:parameterScope()>); separator="\n">

<endif>
<endif>

/* For use in tree output where we are accumulating rule labels via label += ruleRef
* we need a function that knows how to free a return scope when the list is destroyed.
* We cannot just use ANTLR3_FREE because in debug tracking mode, this is a macro.
*/
static void ANTLR3_CDECL freeScope(void * scope)
{
    ANTLR3_FREE(scope);
}

/** \brief Name of the grammar file that generated this code
*/
static const char fileName[] = "<fileName>";

```



```

/** \brief Return the name of the grammar file that generated this code.
*/
static const char * getGrammarFileName()
{
    return fileName;
}

/** \brief Create a new <name> parser and return a context for it.
*
* \param[in] instream Pointer to an input stream interface.
*
* \return Pointer to new parser context upon success.
*/
ANTLR3_API p<name>
<name>New (<inputStreamType> instream<grammar.delegators:{g|, p<g.recognizerName>
<g.delegateName(>>>)
{
    // See if we can create a new parser with the standard constructor
    //
    return <name>NewSSD(instream, NULL<grammar.delegators:{g|, <g.delegateName(>>>));
}

/** \brief Create a new <name> parser and return a context for it.
*
* \param[in] instream Pointer to an input stream interface.
*
* \return Pointer to new parser context upon success.
*/
ANTLR3_API p<name>
<name>NewSSD (<inputStreamType> instream, pANTLR3_RECOGNIZER_SHARED_STATE
state<grammar.delegators:{g|, p<g.recognizerName> <g.delegateName(>>>)
{
    p<name> ctx;    /* Context structure we will build and return */

    ctx = (p<name>) ANTLR3_CALLOC(1, sizeof(<name>));

    if (ctx == NULL)
    {
        // Failed to allocate memory for parser context
        //
        return NULL;
    }

    /* -----
    * Memory for basic structure is allocated, now to fill in
    * the base ANTLR3 structures. We initialize the function pointers
    * for the standard ANTLR3 parser function set, but upon return
    * from here, the programmer may set the pointers to provide custom
    * implementations of each function.

```

```

*
* We don't use the macros defined in <name>.h here, in order that you can get a sense
* of what goes where.
*/

<if(PARSER)>
  /* Create a base parser/recognizer, using the supplied token stream
  */
  ctx->pParser = antlr3ParserNewStream(ANTLR3_SIZE_HINT, instream->tstream, state);
<endif>
<if(TREE_PARSER)>
  /* Create a base Tree parser/recognizer, using the supplied tree node stream
  */
  ctx->pTreeParser = antlr3TreeParserNewStream(ANTLR3_SIZE_HINT, instream, state);
<endif>

  /* Install the implementation of our <name> interface
  */
  <rules:{r | ctx-><r.ruleDescriptor.name> = <r.ruleDescriptor.name>;}; separator="\n">
<if(grammar.delegatedRules)>
  // Install the delegated methods so that they appear to be a part of this
  // parser
  //
  <grammar.delegatedRules:{ruleDescriptor | ctx-><ruleDescriptor.name> = <ruleDescriptor.name>;};
separator="\n">
<endif>

  ctx->free = <name>Free;
  ctx->reset = <name>Reset;
  ctx->getGrammarFileName = getGrammarFileName;

  /* Install the scope pushing methods.
  */
  <rules: {r |<if(r.ruleDescriptor.ruleScope)>
<ruleAttributeScope(scope=r.ruleDescriptor.ruleScope)><\n>
<endif>}>
  <recognizer.scopes:{it |<if(it.isDynamicGlobalScope)>
<globalAttributeScope(it)><\n>
<endif>}>
  <@apifuncs>

  <@end>
<if(grammar.directDelegates)>
  // Initialize the parsers that we are going to delegate some
  // functions to.
  //
  <grammar.directDelegates:
    {g|ctx-><g.delegateName()> = <g.recognizerName>NewSSD(instream, PSRSTATE,

```

```

ctx<grammar.delegators:{g|,<g:delegateName()>>}; separator="\n">
<endif>
<if(grammar.delegators)>
// Install the pointers back to parsers that will delegate us to perform certain functions
// for them.
//
<grammar.delegators:
    {g|ctx-><g:delegateName()> = <g:delegateName()>; separator="\n">
<endif>
    <actions.parser.apifuncs>
    <actions.treeparser.apifuncs>
<if(memoize)>
<if(grammar.grammarIsRoot)>
    /* Create a LIST for recording rule memos.
    */
    RULEMEMO = antlr3IntTrieNew(15); /* 16 bit depth is enough for 32768 rules! */<\n>
<endif>
<endif>
    /* Install the token table
    */
    PSRSTATE->tokenNames = <grammar.composite.rootGrammar.recognizerName>TokenNames;

<@debugStuff()>

    /* Return the newly built parser to the caller
    */
    return ctx;
}

static void
<name>Reset (p<name> ctx)
{
    RECOGNIZER->reset(RECOGNIZER);
}

/** Free the parser resources
*/
static void
<name>Free(p<name> ctx)
{
    /* Free any scope memory
    */
    <rules: {r
|<if(r.ruleDescriptor.ruleScope)><ruleAttributeScopeFree(scope=r.ruleDescriptor.ruleScope)><\n><endif>}>
    <recognizer.scopes:{it |<if(it.isDynamicGlobalScope)><globalAttributeScopeFree(it)><\n><endif>}>

<@cleanup>
<@end>

```

```

<if(grammar.directDelegates)>
// Free the parsers that we delegated to
// functions to.NULL the state so we only free it once.
//
<grammar.directDelegates:
    { g| ctx-><g:delegateName()-><if(TREE_PARSER)>pTreeParser<else>pParser<endif->rec->state = NULL;
      ctx-><g:delegateName()->free(ctx-><g:delegateName()->);}; separator="\n">
<endif>
<if(memoize)>
<if(grammar.grammarIsRoot)>
if (RULEMEMO != NULL)
{
    RULEMEMO->free(RULEMEMO);
    RULEMEMO = NULL;
}
<endif>
<endif>
// Free this parser
//
<if(TREE_PARSER)>
    ctx->pTreeParser->free(ctx->pTreeParser);<\n>
<else>
    ctx->pParser->free(ctx->pParser);<\n>
<endif>

    ANTLR3_FREE(ctx);

    /* Everything is released, so we can return
    */
    return;
}

/** Return token names used by this <grammarType()>
 *
 * The returned pointer is used as an index into the token names table (using the token
 * number as the index).
 *
 * \return Pointer to first char * in the table.
 */
static pANTLR3_UINT8 *getTokenNames()
{
    return <grammar.composite.rootGrammar.recognizerName>TokenNames;
}

<members>

/* Declare the bitsets
*/

```

```
<bitsets:{it | <bitsetDeclare(name={FOLLOW_<it.name>_in_<it.inName><it.tokenIndex>},
    words64=it.bits)>>>
```

```
<if(cyclicDFAs)>
```

```
/* =====
 * DFA tables for the parser
 */
<cyclicDFAs:cyclicDFA() <! dump tables for all DFA !>
/* =====
 * End of DFA tables for the parser
 */
<endif>
```

```
/* =====
 * Parsing rules
 */
<rules; separator="\n\n">
<if(grammar.delegatedRules)>
// Delegated methods that appear to be a part of this
// parser
//
<grammar.delegatedRules:{ruleDescriptor|
    <returnType()> <ruleDescriptor.name>(p<name> ctx<if(ruleDescriptor.parameterScope.attributes)>,
<endif><ruleDescriptor.parameterScope:parameterScope()>)
    \{
        <if(ruleDescriptor.hasReturnValue)>return <endif>ctx-><ruleDescriptor.grammar:delegateName()>-
    <ruleDescriptor.name>(ctx-
    ><ruleDescriptor.grammar:delegateName()><if(ruleDescriptor.parameterScope.attributes)>,
<endif><ruleDescriptor.parameterScope.attributes:{a|<a.name>}; separator=", ">);
    \}}; separator="\n\n">
```

```
<endif>
```

```
/* End of parsing rules
```

```
* =====
*/
```

```
/* =====
 * Syntactic predicates
 */
<synpreds:{p | <synpred(predname=p)>>
/* End of syntactic predicates
 * =====
*/
```

>>

```
parser( grammar,  
  name,  
  scopes,  
  tokens,  
  tokenNames,  
  rules,  
  numRules,  
  bitsets,  
  ASTLabelType,  
  superClass="Parser",  
  labelType="pANTLR3_COMMON_TOKEN",  
  members={<actions.parser.members>}  
  ) ::= <<  
<genericParser(inputStreamType="pANTLR3_COMMON_TOKEN_STREAM", rewriteElementType="TOKEN",  
  ...)>  
>>
```

```
/** How to generate a tree parser; same as parser except the input  
 * stream is a different type.  
 */
```

```
treeParser( grammar,  
  name,  
  scopes,  
  tokens,  
  tokenNames,  
  globalAction,  
  rules,  
  numRules,  
  bitsets,  
  filterMode,  
  labelType={<ASTLabelType>},  
  ASTLabelType="pANTLR3_BASE_TREE",  
  superClass="TreeParser",  
  members={<actions.treeparser.members>}  
  ) ::= <<  
<genericParser(inputStreamType="pANTLR3_COMMON_TREE_NODE_STREAM",  
  rewriteElementType="NODE", ...)>  
>>
```

```
/** A simpler version of a rule template that is specific to the imaginary  
 * rules created for syntactic predicates. As they never have return values  
 * nor parameters etc..., just give simplest possible method. Don't do  
 * any of the normal memoization stuff in here either; it's a waste.  
 * As predicates cannot be inlined into the invoking rule, they need to
```

```

* be in a rule by themselves.
*/
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
// $ANTLR start <ruleName>
static void <ruleName>_fragment(p<name> ctx <ruleDescriptor.parameterScope:parameterScope(>
{
<ruleLabelDefs(>
<ruleLabelInitializations(>
<if(trace)>
    ANTLR3_PRINTF("enter <ruleName> %d failed = %d, backtracking = %d\n",LT(1),failed,BACKTRACKING);
    <block>
    ANTLR3_PRINTF("exit <ruleName> %d, failed = %d, backtracking = %d\n",LT(1),failed,BACKTRACKING);

<else>
    <block>
<endif>
<ruleCleanUp(>
}
// $ANTLR end <ruleName>
>>

synpred(predname) ::= <<
static ANTLR3_BOOLEAN <predname>(p<name> ctx)
{
    ANTLR3_MARKER start;
    ANTLR3_BOOLEAN success;

    BACKTRACKING++;
    <@start(>
    start = MARK();
    <predname>_fragment(ctx); // can never throw exception
    success = !(FAILEDFLAG);
    REWIND(start);
    <@stop(>
    BACKTRACKING--;
    FAILEDFLAG = ANTLR3_FALSE;
    return success;
}<\n>
>>

lexerSynpred(predname) ::= <<
<synpred(predname)>
>>

ruleMemoization(rname) ::= <<
<if(memoize)>
if ( ( BACKTRACKING>0) && (HAVEPARSEDRULE(<ruleDescriptor.index>))) )

```

```

{
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!ruleDescriptor.isSynPred)>
    retval.start = 0;<\n>
<endif>
<endif>
    <(ruleDescriptor.actions.after):execAfter()>
    <finalCode(finalBlock=finally)>
<if(!ruleDescriptor.isSynPred)>
    <scopeClean()><\n>
<endif>
    return <ruleReturnValue()>;
}
<endif>
>>

/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
if (HASEXCEPTION())
{
    goto rule<ruleDescriptor.name>Ex;
}
<if(backtracking)>
if (HASFAILED())
{
    <scopeClean()>
    <@debugClean()>
    return <ruleReturnValue()>;
}
<endif>
>>

/** This rule has failed, exit indicating failure during backtrack */
ruleBacktrackFailure() ::= <<
<if(backtracking)>
if (BACKTRACKING>0)
{
    FAILEDFLAG = <>true_value()>;
    <scopeClean()>
    return <ruleReturnValue()>;
}
<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<

```



```

/**
 * $ANTLR start <ruleName>
 * <fileName>:<description>
 */
static <returnType>
<ruleName>(p<name> ctx<if(ruleDescriptor.parameterScope)>,
<endif><ruleDescriptor.parameterScope:parameterScope()>)
{
    <if(trace)>ANTLR3_PRINTF("enter <ruleName> %s failed=%d, backtracking=%d\n", LT(1),
BACKTRACKING);<endif>
    <ruleDeclarations()>
    <ruleDescriptor.actions.declarations>
    <ruleLabelDefs()>
    <ruleInitializations()>
    <ruleDescriptor.actions.init>
    <ruleMemoization(rname=ruleName)>
    <ruleLabelInitializations()>
    <@preamble()>
    {
        <block>
    }

    <ruleCleanUp()>
<if(exceptions)>
    if (HASEXCEPTION())
    {
<exceptions: {e|<catch(decl=e.decl,action=e.action)><\n>}>
    }
    else
    {
<(ruleDescriptor.actions.after):execAfter()>
    }
<else>
<if(!emptyRule)>
    <if(actions.(actionScope).rulecatch)>
        <actions.(actionScope).rulecatch>
    <else>
        if (HASEXCEPTION())
        {
            PREPORTERROR();
            PRECOVER();
            <@setErrorReturnValue()>
        }
        <if(ruleDescriptor.actions.after)>
    else
    {
        <(ruleDescriptor.actions.after):execAfter()>
    }<\n>

```

```

        <endif>
    <endif>
<endif>
<endif>

    <if(trace)>ANTLR3_PRINTF("exit <ruleName> %d failed=%s backtracking=%s\n", LT(1), failed,
BACKTRACKING);<endif>
    <memoize()>
<if(finally)>
    <finalCode(finalBlock=finally)>
<endif>
    <scopeClean()>
    <@postamble()>
    return <ruleReturnValue()>;
}
/* $ANTLR end <ruleName> */
>>

finalCode(finalBlock) ::= <<
{
    <finalBlock>
}

>>

catch(decl,action) ::= <<
/* catch(decl,action)
*/
{
    <e.action>
}

>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType()> retval;<\n>
<else>
<ruleDescriptor.returnScope.attributes:{ a |
<a.type> <a.name>;
}>
<endif>
<if(memoize)>
ANTLR3_MARKER <ruleDescriptor.name>_StartIndex;
<endif>
>>

ruleInitializations() ::= <<

```

```

/* Initialize rule variables
*/
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.returnScope.attributes:{ a |
<if(a.initValue)>retval.<a.name> = <a.initValue>;<endif>
}>
<else>
<ruleDescriptor.returnScope.attributes:{ a |
<if(a.initValue)><a.name> = <a.initValue>;<endif>
}>
<endif>
<if(memoize)>
<ruleDescriptor.name>_startIndex = INDEX();<\n>
<endif>
<ruleDescriptor.useScopes:{ it |<scopeTop(it)> = <scopePush(it)>;}; separator="\n">
<ruleDescriptor.ruleScope:{ it |<scopeTop(it.name)> = <scopePush(it.name)>;}; separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels]
: {it |<labelType> <it.label.text>;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]
: {it |pANTLR3_VECTOR list_<it.label.text>;}; separator="\n"
>
<[ruleDescriptor.ruleLabels,ruleDescriptor.ruleListLabels]
: ruleLabelDef(); separator="\n"
>
>>

ruleLabelInitializations() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels]
: {it |<it.label.text> = NULL;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,ruleDescriptor.ruleListLabels]
: {it |list_<it.label.text> = NULL;}; separator="\n"
>
<[ruleDescriptor.ruleLabels,ruleDescriptor.ruleListLabels]
: ruleLabelInitVal(); separator="\n"
>
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!ruleDescriptor.isSynPred)>
retval.start = LT(1); retval.stop = retval.start;<\n>
<endif>
<endif>
>>

lexerRuleLabelDefs() ::= <<

```

```

<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it |<labelType> <it.label.text>;}; separator="\n"
>
<ruleDescriptor.charLabels: {it |ANTLR3_UINT32 <it.label.text>;}; separator="\n">
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it |pANTLR3_INT_TRIE list_<it.label.text>;}; separator="\n"
>
>>

```

```

lexerRuleLabelInit() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it |<it.label.text> = NULL;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it |list_<it.label.text> = antlr3IntTrieNew(31);}; separator="\n"
>
>>

```

```

lexerRuleLabelFree() ::= <<
<[ruleDescriptor.tokenLabels,
ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleLabels]
: {it |<it.label.text> = NULL;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,
ruleDescriptor.ruleListLabels,
ruleDescriptor.ruleListLabels]
: {it |list_<it.label.text>->free(list_<it.label.text>);}; separator="\n"
>
>>

```

```

ruleReturnValue() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnName>
<else>
retval
<endif>
<endif>

```

```

<endif>
%>

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if ( BACKTRACKING>0 ) { MEMOIZE(<ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }
<endif>
<endif>
>>

ruleCleanUp() ::= <<

// This is where rules clean up and exit
//
goto rule<ruleDescriptor.name>Ex; /* Prevent compiler warnings */
rule<ruleDescriptor.name>Ex; ;
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
<if(!ruleDescriptor.isSynPred)>
retval.stop = LT(-1);<\n>
<endif>
<endif>
<endif>
>>

scopeClean() ::= <<
<ruleDescriptor.useScopes:{it |<scopePop(it)>}; separator="\n">
<ruleDescriptor.ruleScope:{it |<scopePop(it.name)>}; separator="\n">

>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules, which do not produce tokens.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<
// Comes from: <block.description>
/** \brief Lexer rule generated by ANTLR3
 *
 * $ANTLR start <ruleName>
 *
 * Looks to match the characters the constitute the token <ruleName>
 * from the attached input stream.
 *
 *
 * \remark
 * - lexer->error == ANTLR3_TRUE if an exception was thrown.
 */
static ANTLR3_INLINE

```

```

void m<ruleName>(p<name> ctx<if(ruleDescriptor.parameterScope)>,
<endif><ruleDescriptor.parameterScope:parameterScope()>
{
  ANTLR3_UINT32 _type;
  <ruleDeclarations()>
  <ruleDescriptor.actions.declarations>
  <lexerRuleLabelDefs()>
  <if(trace)>System.out.println("enter <ruleName> "+(char)LA(1)+"
line="+GETLINE()+": "+GETCHARPOSITIONINLINE()+" failed="+failed+"
backtracking="+BACKTRACKING);<endif>

<if(nakedBlock)>
  <ruleMemoization(rname=ruleName)>
  <lexerRuleLabelInit()>
  <ruleDescriptor.actions.init>

  <block><\n>
<else>
  <ruleMemoization(rname=ruleName)>
  <lexerRuleLabelInit()>
  _type = <ruleName>;

  <ruleDescriptor.actions.init>

  <block>
  LEXSTATE->type = _type;
<endif>
  <if(trace)> ANTLR3_FPRINTF(stderr, "exit <ruleName> '%c' line=%d:%d failed = %d, backtracking
=%d\n",LA(1),GETLINE(),GETCHARPOSITIONINLINE(),failed,BACKTRACKING);<endif>
  <ruleCleanUp()>
  <lexerRuleLabelFree()>
  <(ruleDescriptor.actions.after):execAfter()>
  <memoize>
}
// $ANTLR end <ruleName>
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<
/** This is the entry point in to the lexer from an object that
 * wants to generate the next token, such as a pCOMMON_TOKEN_STREAM
 */
static void
mTokens(p<name> ctx)
{
  <block><\n>

```

```

    goto ruleTokensEx; /* Prevent compiler warnings */
ruleTokensEx: ;
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<

// <fileName>:<description>
{
    int alt<decisionNumber>=<maxAlt>;
    <decls>
    <@predecision()>
    <decision>
    <@postdecision()>
    <@prebranch()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)> }>
    }
    <@postbranch()>
}
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
{
    // <fileName>:<description>

    ANTLR3_UINT32 alt<decisionNumber>;

    alt<decisionNumber>=<maxAlt>;

    <decls>
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)> }>
    }
}
>>

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<

```

```

// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
<alts>
<@postalt()>
>>

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
{
  int cnt<decisionNumber>=0;
  <decls>
  <@preloop()>

  for (;;)
  {
    int alt<decisionNumber>=<maxAlt>;
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
      <alts:{a | <altSwitchCase(i,a)>}>
      default:

        if ( cnt<decisionNumber> >= 1 )
        {
          goto loop<decisionNumber>;
        }
        <ruleBacktrackFailure()>
        <earlyExitEx()>
        <@earlyExitException()>
        goto rule<ruleDescriptor.name>Ex;
      }
      cnt<decisionNumber>++;
    }
    loop<decisionNumber>; /* Jump to here if this rule does not match */
  }
}

```



```

    <@postloop()>
}
>>

earlyExitEx() ::= <<
/* mismatchedSetEx()
*/
CONSTRUCTEX();
EXCEPTION->type = ANTLR3_EARLY_EXIT_EXCEPTION;
EXCEPTION->name = (void *)ANTLR3_EARLY_EXIT_NAME;
<\n>
>>
positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<

// <fileName>:<description>
<decls>

<@preloop()>
for (;;)
{
    int alt<decisionNumber>=<maxAlt>;
    <@predecision()>
    <decision>
    <@postdecision()>
    switch (alt<decisionNumber>)
    {
    <alts:{ a | <altSwitchCase(i,a)>}>
    default:
        goto loop<decisionNumber>; /* break out of the loop */
        break;
    }
}
loop<decisionNumber>; /* Jump out to here if this rule does not match */
<@postloop()>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by antlr before code generation
* so we can just use the normal block template
*/
optionalBlock ::= block

optionalBlockSingleAlt ::= block

```

```

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */
altSwitchCase(altNum,alt) ::= <<
case <altNum>:
  <@prealt()>
  <alt>
  break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
  <@declarations()>
  <@initializations()>
  <elements:element()>
  <rew>
  <@cleanup()>
}
>>

// ELEMENTS
/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

/** Dump the elements one per line */
element(e) ::= <<
<@prematch()>
<e.el><\n>
>>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions) ::= <<
<if(label)><label> = (<labelType>)<endif> MATCHT(<token>,
&FOLLOW_<token>_in_<ruleName><elementIndex>);
<checkRuleBacktrackFailure()>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRef(...)>
<listLabel(elem=label,...)>
>>

```

```

listLabel(label,elem) ::= <<
if (list_<label> == NULL)
{
    list_<label>=ctx->vectors->newVector(ctx->vectors);
}
list_<label>->add(list_<label>, <elem>, NULL);
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = LA(1);<\n>
<endif>
MATCHC(<char>);
<checkRuleBacktrackFailure()>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = LA(1);<\n>
<endif>
MATCHRANGE(<a>, <b>);
<checkRuleBacktrackFailure()>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,terminalOptions,postmatchCode="") ::= <<
<if(label)>
<if(LEXER)>
<label>= LA(1);<\n>
<else>
<label>=(<labelType>)LT(1);<\n>
<endif>
<endif>
if ( <s> )
{
    CONSUME();
    <postmatchCode>
<if(!LEXER)>
    PERRORRECOVERY=ANTLR3_FALSE;
<endif>
    <if(backtracking)>FAILEDFLAG=ANTLR3_FALSE;<\n><endif>
}
else
{

```

```

    <ruleBacktrackFailure()>
    <mismatchedSetEx()>
    <@mismatchedSetException()>
<if(LEXER)>
    LRECOVER();
<else>
    RECOVERFROMMISMATCHEDSET(&FOLLOW_set_in_<ruleName><elementIndex>);
<endif>
    goto rule<ruleDescriptor.name>Ex;
}<\n>
>>

```

```

mismatchedSetEx() ::= <<
CONSTRUCTEX();
EXCEPTION->type      = ANTLR3_MISMATCHED_SET_EXCEPTION;
EXCEPTION->name      = (void *)ANTLR3_MISMATCHED_SET_NAME;
<if(PARSER)>
EXCEPTION->expectingSet = &FOLLOW_set_in_<ruleName><elementIndex>;
<endif>
>>

```

```

matchRuleBlockSet ::= matchSet

```

```

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabel(elem=label,...)>
>>

```

```

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <<
<if(label)>
<label>Start = GETCHARINDEX();
MATCHS(<string>);
<checkRuleBacktrackFailure()>
<label> = LEXSTATE->tokFactory->newToken(LEXSTATE->tokFactory);
<label>->setType(<label>, ANTLR3_TOKEN_INVALID);
<label>->setStartIndex(<label>, <label>Start);
<label>->setStopIndex(<label>, GETCHARINDEX()-1);
<label>->input = INPUT->tstream->istream;
<else>
MATCHS(<string>);
<checkRuleBacktrackFailure()><\n>
<endif>
>>

```

```

wildcard(token,label,elementIndex,terminalOptions) ::= <<
<if(label)>
<label>=(<labelType>)LT(1);<\n>

```

```

<endif>
MATCHANYT();
<checkRuleBacktrackFailure()>
>>

wildcardAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<wildcard(...)>
<listLabel(elem=label,...)>
>>

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = LA(1);<\n>
<endif>
MATCHANY();
<checkRuleBacktrackFailure()>
>>

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabel(elem=label,...)>
>>

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
FOLLOWPUSH(FOLLOW_<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)>ctx-><scope:delegateName()-><endif><rule.name>(ctx<if(scope)->
-><scope:delegateName()-><endif><if(args)>, <args; separator=","><endif>);<\n>
FOLLOWPOP();
<checkRuleBacktrackFailure()>
>>

/** ids+=1 */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabel(elem=label,...)>
>>

/** A lexer rule reference
 * The 'rule' argument was the target rule name, but now
 * is type Rule, whose toString is same: the rule name.
 * Now though you can access full rule descriptor stuff.

```

```

*/
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <<
/* <description> */
<if(label)>
{
    ANTLR3_MARKER <label>Start<elementIndex> = GETCHARINDEX();
    <if(scope)>ctx-><scope:delegateName()-><endif>m<rule.name>(ctx<if(scope)>-
><scope:delegateName()-><endif> <if(args)>, <endif><args; separator=", ">;
    <checkRuleBacktrackFailure()>
    <label> = LEXSTATE->tokFactory->newToken(LEXSTATE->tokFactory);
    <label>->setType(<label>, ANTLR3_TOKEN_INVALID);
    <label>->setStartIndex(<label>, <label>Start<elementIndex>);
    <label>->setStopIndex(<label>, GETCHARINDEX()-1);
    <label>->input = INPUT;
}
<else>
<if(scope)>ctx-><scope:delegateName()-><endif>m<rule.name>(ctx<if(scope)>-
><scope:delegateName()-><endif> <if(args)>, <endif><args; separator=", ">;
<checkRuleBacktrackFailure()>
<endif>
>>

/** i+=INT in lexer */
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
<lexerRuleRef(...)>
<listLabel(elem=label,...)>
>>

/** EOF in the lexer */
lexerMatchEOF(label,elementIndex) ::= <<
<if(label)>
{
    ANTLR3_UINT32 <label>Start<elementIndex>;
    <labelType> <label>;
    <label>Start<elementIndex> = GETCHARINDEX();
    MATCHC(ANTLR3_CHARSTREAM_EOF);
    <checkRuleBacktrackFailure()>
    <label> = LEXSTATE->tokFactory->newToken(LEXSTATE->tokFactory);
    <label>->setType(<label>, ANTLR3_TOKEN_EOF);
    <label>->setStartIndex(<label>, <label>Start<elementIndex>);
    <label>->setStopIndex(<label>, GETCHARINDEX()-1);
    <label>->input = INPUT->tstream->istream;
}
<else>
    MATCHC(ANTLR3_CHARSTREAM_EOF);
    <checkRuleBacktrackFailure()>
    <endif>
>>

```

```

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg(>"
recRuleArg()            ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg(> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name)    ::= "$<name>=$<src>.<name>;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList, enclosingTreeLevel, treeLevel) ::= <<
<root:element(>
<actionsAfterRoot:element(>
<if(nullableChildList)>
if ( LA(1)==ANTLR3_TOKEN_DOWN ) {
    MATCHT(ANTLR3_TOKEN_DOWN, NULL);
    <checkRuleBacktrackFailure(>
    <children:element(>
    MATCHT(ANTLR3_TOKEN_UP, NULL);
    <checkRuleBacktrackFailure(>
}
<else>
MATCHT(ANTLR3_TOKEN_DOWN, NULL);
<checkRuleBacktrackFailure(>
<children:element(>
MATCHT(ANTLR3_TOKEN_UP, NULL);
<checkRuleBacktrackFailure(>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
* also hoisted into a prediction expression).
*/
validateSemanticPredicate(pred,description) ::= <<
if ( !(<evalPredicate(...)> )
{
    <ruleBacktrackFailure(>
    <newFPE(...)>
}
>>

newFPE() ::= <<
    CONSTRUCTEX();
    EXCEPTION->type      = ANTLR3_FAILED_PREDICATE_EXCEPTION;
    EXCEPTION->message   = (void *)"<description>";
    EXCEPTION->ruleName = (void *)"<ruleName>";
    <\n>
>>

```

```
// F i x e d D F A (if-then-else)
```

```
dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
```

```
{
  int LA<decisionNumber>_<stateNumber> = LA(<k>);
  <edges; separator="\nelse ">
  else
  {
<if(eotPredictsAlt)>
    alt<decisionNumber>=<eotPredictsAlt>;
<else>
    <ruleBacktrackFailure()>

    <newNVException()>
    goto rule<ruleDescriptor.name>Ex;

<endif>
  }
}
>>
```

```
newNVException() ::= <<
CONSTRUCTEX();
EXCEPTION->type      = ANTLR3_NO_VIABLE_ALT_EXCEPTION;
EXCEPTION->message   = (void *)"<description>";
EXCEPTION->decisionNum = <decisionNumber>;
EXCEPTION->state     = <stateNumber>;
<@noViableAltException()>
<\n>
>>
```

```
/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
```

```
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
{
  int LA<decisionNumber>_<stateNumber> = LA(<k>);
  <edges; separator="\nelse ">
}
>>
```

```
/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
```



```

* anything other than 'a' predicts exiting.
*/

dfaLoopbackStateDecls() ::= <<
ANTLR3_UINT32 LA<decisionNumber>_<stateNumber>;
>>

dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
{
/* dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState)
*/
int LA<decisionNumber>_<stateNumber> = LA(<k>);
<edges; separator="\nelse "><<\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber>=<eotPredictsAlt>; <! if no edges, don't gen ELSE !>
<else>
else
{
alt<decisionNumber>=<eotPredictsAlt>;
}<\n>
<endif>
<endif>
}
>>

/** An accept state indicates a unique alternative has been predicted */
dfaAcceptState(alt) ::= "alt<decisionNumber>=<alt>";

/** A simple edge with an expression. If the expression is satisfied,
* enter to the target state. To handle gated productions, we may
* have to evaluate some predicates for this edge.
*/
dfaEdge(labelExpr, targetState, predicates) ::= <<
if ( <if(predicates)><predicates> && <endif><(<labelExpr>))
{
<targetState>
}
>>

// F i x e d D F A (switch case)

/** A DFA state where a SWITCH may be generated. The code generator
* decides if this is possible: CodeGenerator.canGenerateSwitch().
*/
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( LA(<k>))
{
<edges; separator="\n">

```

```

default:
<if(eotPredictsAlt)>
  alt<decisionNumber>=<eotPredictsAlt>;
<else>
  <ruleBacktrackFailure()>
  <newNVException()>
  goto rule<ruleDescriptor.name>Ex;<\n>
<endif>
}<\n>
>>

dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( LA(<k> )
{
  <edges; separator="\n">
}<\n>
>>

dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
switch ( LA(<k> )
{
<edges; separator="\n"><\n>
<if(eotPredictsAlt)>
default:
  alt<decisionNumber>=<eotPredictsAlt>;
  break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels:{it |case <it>:}; separator="\n">
{
  <targetState>
}
  break;
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
alt<decisionNumber> = cdfa<decisionNumber>.predict(ctx, RECOGNIZER, ISTREAM,
&cdfa<decisionNumber>);

```

```

<checkRuleBacktrackFailure()>
>>

/* Dump DFA tables as static initialized arrays of shorts(16 bits)/characters(8 bits)
 * which are then used to statically initialize the dfa structure, which means that there
 * is no runtime initialization whatsoever, other than anything the C compiler might
 * need to generate. In general the C compiler will lay out memory such that there is no
 * runtime code required.
 */
cyclicDFA(dfa) ::= <<
/** Static dfa state tables for Cyclic dfa:
 * <dfa.description>
 */
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_eot[<dfa.numberofStates>] =
{
<dfa.eot; wrap="\n", separator=", ", null="-1">
};
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_eof[<dfa.numberofStates>] =
{
<dfa.eof; wrap="\n", separator=", ", null="-1">
};
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_min[<dfa.numberofStates>] =
{
<dfa.min; wrap="\n", separator=", ", null="-1">
};
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_max[<dfa.numberofStates>] =
{
<dfa.max; wrap="\n", separator=", ", null="-1">
};
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_accept[<dfa.numberofStates>] =
{
<dfa.accept; wrap="\n", separator=", ", null="-1">
};
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_special[<dfa.numberofStates>] =
{
<dfa.special; wrap="\n", separator=", ", null="-1">
};

/** Used when there is no transition table entry for a particular state */
#define dfa<dfa.decisionNumber>_T_empty NULL

<dfa.edgeTransitionClassMap.keys:{ table |
static const ANTLR3_INT32 dfa<dfa.decisionNumber>_T<i0>[] =
{
<table; separator=", ", wrap="\n", null="-1">
\};}; null = "">

/* Transition tables are a table of sub tables, with some tables

```

```

* reused for efficiency.
*/
static const ANTLR3_INT32 * const dfa<dfa.decisionNumber>_transitions[] =
{
  <dfa.transitionEdgeTables:{ xref[dfa<dfa.decisionNumber>_T<xref>}]; separator=" ", wrap="\n", null="_empty">
};

<if(dfa.specialStateSTs)>
static ANTLR3_INT32 dfa<dfa.decisionNumber>_sst(p<name> ctx, pANTLR3_BASE_RECOGNIZER recognizer,
pANTLR3_INT_STREAM is, pANTLR3_CYCLIC_DFA dfa, ANTLR3_INT32 s)
{
  ANTLR3_INT32  _s;

  _s = s;
  switch (s)
  {
    <dfa.specialStateSTs:{ state |
    case <i0>:

    <state> }; separator="\n">
  }
  <if(backtracking)>
    if (BACKTRACKING > 0)
    {
      FAILEDFLAG = ANTLR3_TRUE;
      return -1;
    }
  <endif>

  CONSTRUCTEX();
  EXCEPTION->type = ANTLR3_NO_VIABLE_ALT_EXCEPTION;
  EXCEPTION->message = (void *)"<dfa.description>";
  EXCEPTION->decisionNum = <dfa.decisionNumber>;
  EXCEPTION->state = _s;
  <@noViableAltException()>
  return -1;
}
<endif>

<@errorMethod()>

/* Declare tracking structure for Cyclic DFA <dfa.decisionNumber>
*/
static
ANTLR3_CYCLIC_DFA cdfa<dfa.decisionNumber>
= {
  <dfa.decisionNumber>, /* Decision number of this dfa */
  /* Which decision this represents: */

```

```

    (const pANTLR3_UCHAR)"<dfa.description>",
<if(dfa.specialStateSTs)>
    (CDFA_SPECIAL_FUNC) dfa<dfa.decisionNumber>_sst,
<else>
    (CDFA_SPECIAL_FUNC) antlr3dfaspecialStateTransition, /* Default special state transition function */
<endif>

    antlr3dfaspecialTransition, /* DFA specialTransition is currently just a default function in the runtime */
    antlr3dfapredict, /* DFA simulator function is in the runtime */
    dfa<dfa.decisionNumber>_eot, /* EOT table */
    dfa<dfa.decisionNumber>_eof, /* EOF table */
    dfa<dfa.decisionNumber>_min, /* Minimum tokens for each state */
    dfa<dfa.decisionNumber>_max, /* Maximum tokens for each state */
    dfa<dfa.decisionNumber>_accept, /* Accept table */
    dfa<dfa.decisionNumber>_special, /* Special transition states */
    dfa<dfa.decisionNumber>_transitions /* Table of transition tables */

};
/* End of Cyclic DFA <dfa.decisionNumber>
* -----
*/
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
* state.
*/
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
{
    ANTLR3_UINT32 LA<decisionNumber>_<stateNumber>;<\n>
    ANTLR3_MARKER index<decisionNumber>_<stateNumber>;<\n>

    LA<decisionNumber>_<stateNumber> = LA(1);<\n>
    <if(semPredState)> <! get next lookahead symbol to test edges, then rewind !>
    index<decisionNumber>_<stateNumber> = INDEX();<\n>
    REWINDLAST();<\n>
    <endif>
    s = -1;
    <edges; separator="\nelse ">
    <if(semPredState)> <! return input cursor to state before we rewound !>
    SEEK(index<decisionNumber>_<stateNumber>);<\n>
    <endif>
    if ( s>=0 )
    {
        return s;
    }
}
break;
>>

```

```

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ( <if(predicates)><predicates> && <endif><labelExpr> )
{
  s = <targetStateNumber>;
}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "( (<left> ) && (<right> ) )"

orPredicates(operands) ::= "(<operands:{o|(<o>)}; separator=\"||\">)"

notPredicate(pred) ::= "!(<evalPredicate(pred,{})>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "<pred>(ctx)"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<stateNumber> == <atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "LA(<k>) == <atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
((LA<decisionNumber>_<stateNumber> >= <lower>) && (LA<decisionNumber>_<stateNumber> \<= <upper>))
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= "((LA(<k>) >= <lower>) &&
(LA(<k>) \<= <upper>))"

setTest(ranges) ::= "<ranges; separator=\" || \">"

// A T T R I B U T E S

```

```

makeScopeSet() ::= <<
/* makeScopeSet()
*/
/** Definition of the <scope.name> scope variable tracking
* structure. An instance of this structure is created by calling
* <name>_<scope.name>Push().
*/
typedef struct <scopeStruct(sname=scope.name,...)>_struct
{
/** Function that the user may provide to be called when the
* scope is destroyed (so you can free pANTLR3_HASH_TABLES and so on)
*
* \param POinter to an instance of this typedef/struct
*/
void (ANTLR3_CDECL *free) (struct <scopeStruct(sname=scope.name,...)>_struct * frame);

/* =====
* Programmer defined variables...
*/
<scope.attributes:{it |<it.decl>;}; separator="\n">

/* End of programmer defined variables
* =====
*/
}
<scopeStruct(sname=scope.name,...)>, * <scopeType(sname=scope.name,...)>;

>>

globalAttributeScopeDecl(scope) ::= <<
<if(scope.attributes)>
/* globalAttributeScopeDecl(scope)
*/
<makeScopeSet(...)>
<endif>
>>

ruleAttributeScopeDecl(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeDecl(scope)
*/
<makeScopeSet(...)>
<endif>
>>

globalAttributeScopeFuncDecl(scope) ::=
<<

```

```

/* globalAttributeScopeFuncDecl(scope)
*/
<if(scope.attributes)>
/* -----
* Function declaration for creating a <name>_<scope.name> scope set
*/
static <scopeType(sname=scope.name,...)> <scopePushName(sname=scope.name,...)>(p<name> ctx);
static void ANTLR3_CDECL <scope.name>Free(<scopeType(sname=scope.name)> scope);
/* ----- */

<endif>
>>

globalAttributeScopeFuncMacro(scope) ::= <<
<if(scope.attributes)>
/* globalAttributeScopeFuncMacro(scope)
*/
/** Function for popping the top value from a <scopeStack(sname=scope.name)>
*/
void
<scopePopName(sname=scope.name,...)>(p<name> ctx)
{
// First see if the user defined a function they want to be called when a
// scope is popped/freed.
//
// If the user supplied the scope entries with a free function,then call it first
//
if (SCOPE_TOP(<scope.name>)->free != NULL)
{
SCOPE_TOP(<scope.name>)->free(SCOPE_TOP(<scope.name>));
}

// Now we decrement the scope's upper limit bound. We do not actually pop the scope as
// we want to reuse scope entries if we do continuous push and pops. Most scopes don't
// next too far so we don't want to keep freeing and allocating them
//
ctx-><scopeStack(sname=scope.name,...)>_limit--;
SCOPE_TOP(<scope.name>) = (<scopeType(sname=scope.name)>)(ctx-><scopeStack(sname=scope.name,...)>-
>get(ctx-><scopeStack(sname=scope.name,...)>, ctx-><scopeStack(sname=scope.name,...)>_limit - 1));
}
<endif>
>>

ruleAttributeScopeFuncDecl(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeFuncDecl(scope)
*/
/* -----

```



```

* Function declarations for creating a <name>_<scope.name> scope set
*/
static <scopeType(sname=scope.name,...)> <scopePushName(sname=scope.name,...)>(p<name> ctx);
static void ANTLR3_CDECL <scope.name>Free(<scopeType(sname=scope.name)> scope);
/* ----- */

<endif>
>>

ruleAttributeScopeFuncMacro(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeFuncMacro(scope)
*/
/** Function for popping the top value from a <scopeStack(sname=scope.name,...)>
*/
void
<scopePopName(sname=scope.name,...)>(p<name> ctx)
{
    // First see if the user defined a function they want to be called when a
    // scope is popped/freed.
    //
    // If the user supplied the scope entries with a free function, then call it first
    //
    if (SCOPE_TOP(<scope.name>)->free != NULL)
    {
        SCOPE_TOP(<scope.name>)->free(SCOPE_TOP(<scope.name>));
    }

    // Now we decrement the scope's upper limit bound. We do not actually pop the scope as
    // we want to reuse scope entries if we do continuous push and pops. Most scopes don't
    // next too far so we don't want to keep freeing and allocating them
    //
    ctx-><scopeStack(sname=scope.name,...)>_limit--;
    SCOPE_TOP(<scope.name>) = (<scopeType(sname=scope.name)>)(ctx-><scopeStack(sname=scope.name,...)>-
>get(ctx-><scopeStack(sname=scope.name,...)>, ctx-><scopeStack(sname=scope.name,...)>_limit - 1));
}

<endif>
>>

globalAttributeScopeDef(scope) ::=
<<
/* globalAttributeScopeDef(scope)
*/
<if(scope.attributes)>
/** Pointer to the <scope.name> stack for use by <scopePushName(sname=scope.name)>()
* and <scopePopName(sname=scope.name,...)>()
*/

```

```

pANTLR3_STACK <scopeStack(sname=scope.name)>;
ANTLR3_UINT32 <scopeStack(sname=scope.name)>_limit;
/** Pointer to the top of the stack for the global scope <scopeStack(sname=scope.name)>
*/
<scopeType(sname=scope.name,...)> (*<scopePushName(sname=scope.name,...)>)(struct <name>_Ctx_struct *
ctx);
<scopeType(sname=scope.name,...)> <scopeTopDecl(sname=scope.name,...)>;

<endif>
>>

ruleAttributeScopeDef(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeDef(scope)
*/
/** Pointer to the <scope.name> stack for use by <scopePushName(sname=scope.name)>()
* and <scopePopName(sname=scope.name,...)>()
*/
pANTLR3_STACK <scopeStack(sname=scope.name,...)>;
ANTLR3_UINT32 <scopeStack(sname=scope.name,...)>_limit;
<scopeType(sname=scope.name,...)> (*<scopePushName(sname=scope.name,...)>)(struct <name>_Ctx_struct *
ctx);
<scopeType(sname=scope.name,...)> <scopeTopDecl(sname=scope.name,...)>;

<endif>
>>

globalAttributeScopeFuncs(scope) ::= <<
<if(scope.attributes)>
/* globalAttributeScopeFuncs(scope)
*/
<attributeFuncs(scope)>
<endif>
>>

ruleAttributeScopeFuncs(scope) ::= <<
<if(scope.attributes)>
/* ruleAttributeScopeFuncs(scope)
*/
<attributeFuncs(scope)>
<endif>
>>

globalAttributeScope(scope) ::= <<
<if(scope.attributes)>
/* globalAttributeScope(scope)
*/
ctx-><scopePushName(sname=scope.name,...)> = <scopePushName(sname=scope.name,...)>;

```

```

ctx-><scopeStack(sname=scope.name,...)> = antlr3StackNew(0);
ctx-><scopeStack(sname=scope.name,...)>_limit = 0;
<scopeTop(sname=scope.name,...)> = NULL;
<endif>
>>

ruleAttributeScope(scope) ::=
<<
<if(scope.attributes)>
/* ruleAttributeScope(scope)
*/
ctx-><scopePushName(sname=scope.name,...)> = <scopePushName(sname=scope.name,...)>;
ctx-><scopeStack(sname=scope.name,...)> = antlr3StackNew(0);
ctx-><scopeStack(sname=scope.name,...)>_limit = 0;
<scopeTop(sname=scope.name,...)> = NULL;
<endif>
>>
globalAttributeScopeFree(scope) ::= <<
<if(scope.attributes)>
/* globalAttributeScope(scope)
*/
ctx-><scopeStack(sname=scope.name,...)>->free(ctx-><scopeStack(sname=scope.name,...)>);
<endif>
>>

ruleAttributeScopeFree(scope) ::=
<<
<if(scope.attributes)>
/* ruleAttributeScope(scope)
*/
ctx-><scopeStack(sname=scope.name,...)>->free(ctx-><scopeStack(sname=scope.name,...)>);
<endif>
>>

scopeTopDecl(sname) ::= <<
p<name>_<sname>Top
>>

scopeTop(sname) ::= <<
ctx-><scopeTopDecl(sname=sname,...)>
>>

scopePop(sname) ::= <<
<scopePopName(sname=sname,...)>(ctx);
>>

scopePush(sname) ::= <<
p<name>_<sname>Push(ctx)

```

>>

```
scopePopName(sname) ::= <<
p<name>_<sname>Pop
>>
```

```
scopePushName(sname) ::= <<
p<name>_<sname>Push
>>
```

```
scopeType(sname) ::= <<
p<name>_<sname>_SCOPE
>>
```

```
scopeStruct(sname) ::= <<
<name>_<sname>_SCOPE
>>
```

```
scopeStack(sname) ::= <<
p<name>_<sname>Stack
>>
```

```
attributeFuncs(scope) ::= <<
<if(scope.attributes)>
/* attributeFuncs(scope)
*/
```

```
static void ANTLR3_CDECL <scope.name>Free(<scopeType(sname=scope.name)> scope)
{
    ANTLR3_FREE(scope);
}
```

```
/** \brief Allocate initial memory for a <name> <scope.name> scope variable stack entry and
*     add it to the top of the stack.
```

```
*
*
```

```
* \remark
```

```
* By default the structure is freed with ANTLR_FREE(), but you can use the
```

```
* the \@init action to install a pointer to a custom free() routine by
```

```
* adding the code:
```

```
* \code
```

```
* <scopeTop(sname=scope.name)>->free = myroutine;
```

```
* \endcode
```

```
*
*
```

```
* With lots of comments of course! The routine should be declared in
```

```
* \@members { } as:
```

```
* \code
```

```
* void ANTLR3_CDECL myfunc( <scopeType(sname=scope.name)> ptr).
```

```
* \endcode
```

```

*
* It should perform any custom freeing stuff that you need (call ANTLR_FREE3, not free())
* NB: It should not free the pointer it is given, which is the scope stack entry itself
* and will be freed by the function that calls your custom free routine.
*
*/
static <scopeType(sname=scope.name)>
<scopePushName(sname=scope.name)>(p<name> ctx)
{
    /* Pointer used to create a new set of attributes
    */
    <scopeType(sname=scope.name)>    newAttributes;

    /* Allocate the memory for a new structure if we need one.
    */
    if (ctx-><scopeStack(sname=scope.name)>->size(ctx-><scopeStack(sname=scope.name)>) > ctx-
    ><scopeStack(sname=scope.name)>_limit)
    {
        // The current limit value was less than the number of scopes available on the stack so
        // we can just reuse one. Our limit tracks the stack count, so the index of the entry we want
        // is one less than that, or conveniently, the current value of limit.
        //
        newAttributes = (<scopeType(sname=scope.name)>)ctx-><scopeStack(sname=scope.name)>->get(ctx-
    ><scopeStack(sname=scope.name)>, ctx-><scopeStack(sname=scope.name)>_limit);
    }
    else
    {
        // Need a new allocation
        //
        newAttributes = (<scopeType(sname=scope.name)>)
    ANTLR3_MALLOC(sizeof(<scopeStruct(sname=scope.name)>));
        if (newAttributes != NULL)
        {
            /* Standard ANTLR3 library implementation
            */
            ctx-><scopeStack(sname=scope.name)>->push(ctx-><scopeStack(sname=scope.name)>, newAttributes,
    (void (*)(void *))<scope.name>Free);
        }
    }

    // Blank out any previous free pointer, the user might or might install a new one.
    //
    newAttributes->free = NULL;

    // Indicate the position in the available stack that the current level is at
    //
    ctx-><scopeStack(sname=scope.name)>_limit++;
}

```

```

/* Return value is the pointer to the new entry, which may be used locally
 * without de-referencing via the context.
 */
return newAttributes;
}<\n>

<endif>
>>
returnStructName(r) ::= "<r.name>_return"

returnType() ::= <%
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.grammar.recognizerName>_<ruleDescriptor:returnStructName()>
<else>
<if(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
<endif>
<else>
ANTLR3_BOOLEAN
<endif>
%>

/** Generate the C type associated with a single or multiple return
 * value(s).
 */
ruleLabelType(referencedRule) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
<referencedRule.grammar.recognizerName>_<referencedRule.name>_return
<else>
<if(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
<endif>
%>

delegateName(d) ::= <<
<if(d.label)><d.label><else>g<it.name><endif>
>>

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "0".
 */

```

```

initValue(typeName) ::= <<
= <<TypeInitMap.(typeName)>
>>

/** Define a rule label */
ruleLabelDef(label) ::= <<
<ruleLabelType(referencedRule=label.referencedRule)> <label.label.text>;
#undef RETURN_TYPE_<label.label.text>
#define RETURN_TYPE_<label.label.text> <ruleLabelType(referencedRule=label.referencedRule)><\n>
>>
/** Rule label default value */
ruleLabelInitVal(label) ::= <<
>>

ASTLabelType() ::=
"<if(recognizer.ASTLabelType)><recognizer.ASTLabelType><else>pANTLR3_BASE_TREE<endif>"

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasMultipleReturnValues)>
typedef struct <ruleDescriptor.grammar.recognizerName>_<ruleDescriptor:returnStructName()>_struct
{
<if(!TREE_PARSER)>
/** Generic return elements for ANTLR3 rules that are not in tree parsers or returning trees
 */
pANTLR3_COMMON_TOKEN start;
pANTLR3_COMMON_TOKEN stop;
<else>
<recognizer.ASTLabelType> start;
<recognizer.ASTLabelType> stop;
<endif>
<@ruleReturnMembers()>
<ruleDescriptor.returnScope.attributes:{it |<it.type> <it.name>;}; separator="\n">
}
<ruleDescriptor.grammar.recognizerName>_<ruleDescriptor:returnStructName()>;<\n><\n>
<endif>
<endif>
>>

parameterScope(scope) ::= <<
<scope.attributes:{it |<it.decl>}; separator=", ">
>>

parameterAttributeRef(attr) ::= "<attr.name>"

```

```

parameterSetAttributeRef(attr,expr) ::= "<attr.name>=<expr>";

/** Note that the scopeAttributeRef does not have access to the
 * grammar name directly
 */
scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
((SCOPE_TYPE(<scope>))(ctx->SCOPE_STACK(<scope>)->get( ctx->SCOPE_STACK(<scope>), ctx-
>SCOPE_STACK(<scope>)->size(ctx->SCOPE_STACK(<scope>)) - <negIndex> - 1 ))-><attr.name>
<else>
<if(index)>
((SCOPE_TYPE(<scope>))(ctx->SCOPE_STACK(<scope>)->get(ctx->SCOPE_STACK(<scope>),
(ANTLR3_UINT32)<index> ) ))-><attr.name>
<else>
(SCOPE_TOP(<scope>))-><attr.name>
<endif>
<endif>
%>

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
((SCOPE_TYPE(<scope>))(ctx->SCOPE_STACK(<scope>)->get( ctx->SCOPE_STACK(<scope>), ctx-
>SCOPE_STACK(<scope>)->size(ctx->SCOPE_STACK(<scope>)) - <negIndex> - 1 ))-><attr.name> = <expr>;
<else>
<if(index)>
((SCOPE_TYPE(<scope>))(ctx->SCOPE_STACK(<scope>)->get(ctx->SCOPE_STACK(<scope>),
(ANTLR3_UINT32)<index> ) ))-><attr.name> = <expr>;
<else>
(SCOPE_TOP(<scope>))-><attr.name>=<expr>;
<endif>
<endif>
%>

/** $x is either global scope or x is rule with dynamic scope; refers
 * to stack itself not top of stack. This is useful for predicates
 * like {$function.size(>0 && $function::name.equals("foo"))?
 */
isolatedDynamicScopeRef(scope) ::= "ctx->SCOPE_STACK(<scope>)"

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <<
<if(referencedRule.hasMultipleReturnValues)>
<scope>.<attr.name>
<else>
<scope>
<endif>
>>

```



```

returnAttributeRef(ruleDescriptor,attr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>
<else>
<attr.name>
<endif>
>>

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name>=<expr>;
<else>
<attr.name>=<expr>;
<endif>
>>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach
//
tokenLabelPropertyRef_text(scope,attr) ::= "(<scope>->getText(<scope>))"
tokenLabelPropertyRef_type(scope,attr) ::= "(<scope>->getType(<scope>))"
tokenLabelPropertyRef_line(scope,attr) ::= "(<scope>->getLine(<scope>))"
tokenLabelPropertyRef_pos(scope,attr) ::= "(<scope>->getCharPositionInLine(<scope>))"
tokenLabelPropertyRef_channel(scope,attr) ::= "(<scope>->getChannel(<scope>))"
tokenLabelPropertyRef_index(scope,attr) ::= "(<scope>->getTokenIndex(<scope>))"
tokenLabelPropertyRef_tree(scope,attr) ::= "(<scope>->tree)"
tokenLabelPropertyRef_int(scope,attr) ::= "(<scope>->getText(<scope>->toInt32(<scope>->getText(<scope>)))"

ruleLabelPropertyRef_start(scope,attr) ::= "<scope>.start"
ruleLabelPropertyRef_stop(scope,attr) ::= "<scope>.stop"
ruleLabelPropertyRef_tree(scope,attr) ::= "<scope>.tree"
ruleLabelPropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
(STRSTREAM->toStringSS(STRSTREAM, <scope>.start, <scope>.start))
<else>
(STRSTREAM->toStringTT(STRSTREAM, <scope>.start, <scope>.stop))
<endif>
>>

ruleLabelPropertyRef_st(scope,attr) ::= "<scope>.st"

/** Isolated $RULE ref ok in lexer as it's a Token */

```

```

lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::= "(<scope>->getType(<scope>))"
lexerRuleLabelPropertyRef_line(scope,attr) ::= "(<scope>->getLine(<scope>))"
lexerRuleLabelPropertyRef_pos(scope,attr) ::= "(<scope>->getCharPositionInLine(<scope>))"
lexerRuleLabelPropertyRef_channel(scope,attr) ::= "(<scope>->getChannel(<scope>))"
lexerRuleLabelPropertyRef_index(scope,attr) ::= "(<scope>->getTokenIndex(<scope>))"
lexerRuleLabelPropertyRef_text(scope,attr) ::= "(<scope>->getText(<scope>))"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.start"
rulePropertyRef_stop(scope,attr) ::= "retval.stop"
rulePropertyRef_tree(scope,attr) ::= "retval.tree"
rulePropertyRef_text(scope,attr) ::= <<
<if(TREE_PARSER)>
INPUT->toStringSS(INPUT, ADAPTOR->getTokenStartIndex(ADAPTOR, retval.start), ADAPTOR-
>getTokenStopIndex(ADAPTOR, retval.start))
<else>
STRSTREAM->toStringTT(STRSTREAM, retval.start, LT(-1))
<endif>
>>
rulePropertyRef_st(scope,attr) ::= "retval.st"

lexerRulePropertyRef_text(scope,attr) ::= "LEXER->getText(LEXER)"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "LEXSTATE->tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "LEXSTATE->tokenStartCharPositionInLine"
lexerRulePropertyRef_channel(scope,attr) ::= "LEXSTATE->channel"
lexerRulePropertyRef_start(scope,attr) ::= "LEXSTATE->tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(LEXER->getCharIndex(LEXER)-1)"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_int(scope,attr) ::= "LEXER->getText(LEXER)->toInt32(LEXER->getText(LEXER))"

// setting $st and $tree is allowed in local rule. everything else is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.tree=<expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.st=<expr>;"

/** How to deal with an @after for C targets. Because we cannot rely on
 * any garbage collection, after code is executed even in backtracking
 * mode. Must be documented clearly.
 */
execAfter(action) ::= <<
{
  <action>
}
>>

```

```

/** How to execute an action (when not backtracking) */
execAction(action) ::= <<
<if(backtracking)>
<if(actions.(actionScope).synpredgate)>
if ( <actions.(actionScope).synpredgate> )
{
    <action>
}
<else>
if ( BACKTRACKING == 0 )
{
    <action>
}
<endif>
<else>
{
    <action>
}
<endif>
>>

// M I S C (properties, etc...)

bitsetDeclare(name, words64) ::= <<

/** Bitset defining follow set for error recovery in rule state: <name> */
static ANTLR3_BITWORD <name>_bits[] = { <words64:{it |ANTLR3_UINT64_LIT(<it>)}; separator=", "> };
static ANTLR3_BITSET_LIST <name> = { <name>_bits, <length(words64)> };
>>

bitset(name, words64) ::= <<
antlr3BitsetSetAPI(&<name>);<\n>
>>

codeFileExtension() ::= ".c"

true_value() ::= "ANTLR3_TRUE"
false_value() ::= "ANTLR3_FALSE"

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/templates/C/C.stg
No license file was found, but licenses were detected in source scan.

/*
* [The "BSD license"]
* Copyright (c) 2011 Terence Parr
* All rights reserved.

```

```

*
* Conversion to C#:
* Copyright (c) 2011 Sam Harwell, Pixel Mine, Inc.
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
/** Template subgroup to add template rewrite output
* If debugging, then you'll also get STDbg.stg loaded.
*/

@outputFile.imports() ::= <<
<@super.imports()>
using Antlr3.ST;
using Antlr3.ST.Language;
>>

@genericParser.members() ::= <<
<@super.members()>
private StringTemplateGroup _templateGroup = new StringTemplateGroup("<name>Templates",
typeof(AngleBracketTemplateLexer) );

public StringTemplateGroup TemplateGroup
{
get { return _templateGroup; }
set { _templateGroup = value; }
}
>>

```

```

ruleReturnBaseType() ::= <%
Template<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope\<StringTemplate, <labelType>>
%>

/** x+=rule when output=template */
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRef(...)>
<listLabelElem(elem={<label>.Template},elemType="StringTemplate",...)>
>>

rewriteTemplate(alts) ::= <<

// TEMPLATE REWRITE
<if(backtracking)>
if (<actions.(actionScope).synpredgate>)
{
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
}
<else>
<alts:rewriteTemplateAlt(); separator="else ">
<if(rewriteMode)><replaceTextInLine()><endif>
<endif>
>>

replaceTextInLine() ::= <<
<if(TREE_PARSER)>
((TokenRewriteStream)input.TokenStream).Replace(
input.TreeAdaptor.GetTokenStartIndex(retval.Start),
input.TreeAdaptor.GetTokenStopIndex(retval.Start),
retval.Template);
<else>
((TokenRewriteStream)input).Replace(
retval.Start.TokenIndex,
input.LT(-1).TokenIndex,
retval.Template);
<endif>
>>

rewriteTemplateAlt(it) ::= <<
// <it.description>
<if(it.pred)>
if (<it.pred>)
{
retval.Template = <it.alt>;
}<\n>
<else>

```

```

{
    retval.Template = <it.alt>;
}
<\n>
<endif>
>>

rewriteEmptyTemplate(alts) ::= <<
null;
>>

/** Invoke a template with a set of attribute name/value pairs.
 * Set the value of the rule's template *after* having set
 * the attributes because the rule's template might be used as
 * an attribute to build a bigger template; you get a self-embedded
 * template.
 */
rewriteExternalTemplate(name,args) ::= <%
TemplateGroup.GetInstanceOf("<name>"<optionalArguments(args)>)
%>

/** expr is a string expression that says what template to load */
rewriteIndirectTemplate(expr,args) ::= <%
TemplateGroup.GetInstanceOf(<expr>"<optionalArguments(args)>"
%>

/** Invoke an inline template with a set of attribute name/value pairs */
rewriteInlineTemplate(args, template) ::= <%
new StringTemplate(TemplateGroup, "<template>"<optionalArguments(args)>)
%>

optionalArguments(args) ::= <<
<if(args)>,
    new Dictionary<string, object>() { <args:optionalArgument(); separator=", "> }
<endif>
>>

optionalArgument(it) ::= <<
{"<it.name>", <it.value>}
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
<action>
>>

/** An action has %st.attrName=expr; or % {st}.attrName=expr; */
actionSetAttribute(st,attrName,expr) ::= <<
(<st>).SetAttribute("<attrName>",<expr>);

```

>>

```
/** Translate %{stringExpr} */  
actionStringConstructor(stringExpr) ::= <<  
new StringTemplate(TemplateGroup,<stringExpr>)  
>>
```

Found in path(s):

```
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-  
jar/org/antlr/codegen/templates/CSharp3/ST.stg  
No license file was found, but licenses were detected in source scan.
```

```
/*  
[The "BSD license"]  
Copyright (c) 2005-2006 Terence Parr  
All rights reserved.
```

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name of the author may not be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

```
*/  
group AST;
```

```
@outputFile.imports() ::= <<  
<@super.imports(>  
>>
```

```
@genericParser.members() ::= <<  
<@super.members(>  
<parserMembers(>
```

```

>>

/** Add an adaptor property that knows how to build trees */
parserMembers() ::= <<
<!protected TreeAdaptor adaptor = new CommonTreeAdaptor();<\n!>
setTreeAdaptor: function(adaptor) {
    this.adaptor = adaptor;
    <grammar.directDelegates: {g|<g:delegateName()>.setTreeAdaptor(this.adaptor);}>
},
getTreeAdaptor: function() {
    return this.adaptor;
},
>>

@returnScope.ruleReturnMembers() ::= <<
getTree: function() { return this.tree; }
>>

/** Add a variable to track rule's return AST */
ruleDeclarations() ::= <<
<super.ruleDeclarations()>
var root_0 = null;<\n>
>>

ruleLabelDefs() ::= <<
<super.ruleLabelDefs()>
<ruleDescriptor.tokenLabels: { var <it.label.text>_tree=null; }; separator="\n">
<ruleDescriptor.tokenListLabels: { var <it.label.text>_tree=null; }; separator="\n">
<ruleDescriptor.allTokenRefsInAltsWithRewrites
    : { var stream_<it>=new org.antlr.runtime.tree.RewriteRuleTokenStream(this.adaptor,"token <it>"); };
separator="\n">
<ruleDescriptor.allRuleRefsInAltsWithRewrites
    : { var stream_<it>=new org.antlr.runtime.tree.RewriteRuleSubtreeStream(this.adaptor,"rule <it>"); };
separator="\n">
>>

/** When doing auto AST construction, we must define some variables;
 * These should be turned off if doing rewrites. This must be a "mode"
 * as a rule could have both rewrite and AST within the same alternative
 * block.
 */
@alt.declarations() ::= <<
<if(autoAST)>
<if(outerAlt)>
<if(!rewriteMode)>
root_0 = this.adaptor.nil();<\n>
<endif>
<endif>

```



```

<endif>
>>

// Tracking Rule Elements

/** ID and track it for use in a rewrite rule */
tokenRefTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)> <! Track implies no auto AST construction!>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** ids+=ID and track it for use in a rewrite rule; adds to ids *and*
 * to the tracking list stream_ID for use in the rewrite.
 */
tokenRefTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefTrack(...)>
<listLabel(elem=label,...)>
>>

/** ^(ID ...) track for rewrite */
tokenRefRuleRootTrack(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefBang(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<token>.add(<label>);<\n>
>>

/** Match ^(label+=TOKEN ...) track for rewrite */
tokenRefRuleRootTrackAndListLabel(token,label,elementIndex,terminalOptions) ::= <<
<tokenRefRuleRootTrack(...)>
<listLabel(elem=label,...)>
>>

wildcardTrack(label,elementIndex) ::= <<
<super.wildcard(...)>
>>

/** rule when output=AST and tracking for rewrite */
ruleRefTrack(rule,label,elementIndex,args,scope) ::= <<
<super.ruleRef(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule.name>.add(<label>.getTree());
>>

/** x+=rule when output=AST and tracking for rewrite */
ruleRefTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefTrack(...)>
<listLabel(elem=label+".getTree()",...)>
>>

/** ^(rule ...) rewrite */

```

```

ruleRefRuleRootTrack(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRoot(...)>
<if(backtracking)>if ( <actions.(actionScope).synpredgate> ) <endif>stream_<rule>.add(<label>.getTree());
>>

/** ^(x+=rule ...) rewrite */
ruleRefRuleRootTrackAndListLabel(rule,label,elementIndex,args,scope) ::= <<
<ruleRefRuleRootTrack(...)>
<listLabel(elem=label+".getTree()",...)>
>>

// R e w r i t e

rewriteCode(
alts, description,
referencedElementsDeep, // ALL referenced elements to right of ->
referencedTokenLabels,
referencedTokenListLabels,
referencedRuleLabels,
referencedRuleListLabels,
referencedWildcardLabels,
referencedWildcardListLabels,
rewriteBlockLevel, enclosingTreeLevel, treeLevel) ::=
<<

// AST REWRITE
// elements: <referencedElementsDeep; separator=", ">
// token labels: <referencedTokenLabels; separator=", ">
// rule labels: <referencedRuleLabels; separator=", ">
// token list labels: <referencedTokenListLabels; separator=", ">
// rule list labels: <referencedRuleListLabels; separator=", ">
<if(backtracking)>
if ( <actions.(actionScope).synpredgate> ) {<n>
<endif>
<prevRuleRootRef(>).tree = root_0;
<rewriteCodeLabels(>
root_0 = this.adaptor.nil();
<alts:rewriteAlt(); separator="else ">
<! if tree parser and rewrite=true !>
<if(TREE_PARSER)>
<if(rewriteMode)>
<prevRuleRootRef(>).tree = this.adaptor.rulePostProcessing(root_0);
this.input.replaceChildren(this.adaptor.getParent(retval.start),
this.adaptor.getChildIndex(retval.start),
this.adaptor.getChildIndex(_last),
retval.tree);

<endif>
<endif>

```

```

<! if parser or tree-parser && rewrite!=true, we need to set result !>
<if(!TREE_PARSER)>
<prevRuleRootRef(>.tree = root_0;
<else>
<if(!rewriteMode)>
<prevRuleRootRef(>.tree = root_0;
<endif>
<endif>
<if(backtracking)>
}
<endif>
>>

rewriteCodeLabels() ::= <<
<referencedTokenLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRule<rewriteElementType>Stream(this.adaptor,"token
<it>",<it>)};
separator="\n"
>
<referencedTokenListLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRule<rewriteElementType>Stream(this.adaptor,"token
<it>",<it>);};
separator="\n"
>
<referencedWildcardLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRuleSubtreeStream(this.adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedWildcardListLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRuleSubtreeStream(this.adaptor,"wildcard <it>",<it>)};
separator="\n"
>
<referencedRuleLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRuleSubtreeStream(this.adaptor,"token
<it>",<it>!=null?<it>.tree:null)};
separator="\n"
>
<referencedRuleListLabels
: {var stream_<it>=new org.antlr.runtime.tree.RewriteRuleSubtreeStream(this.adaptor,"token <it>",<it>)};
separator="\n"
>
>>

/** Generate code for an optional rewrite block; note it uses the deep ref'd element
 * list rather shallow like other blocks.
 */
rewriteOptionalBlock(
alt,rewriteBlockLevel,

```

```

referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
if ( <referencedElementsDeep:{el | stream_<el>.hasNext()}; separator="||"> ) {
    <alt>
}
<referencedElementsDeep:{el | stream_<el>.reset();<\n>}>
>>

```

```

rewriteClosureBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
// <fileName>:<description>
while ( <referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) {
    <alt>
}
<referencedElements:{el | stream_<el>.reset();<\n>}>
>>

```

```

rewritePositiveClosureBlock(
alt,rewriteBlockLevel,
referencedElementsDeep, // all nested refs
referencedElements, // elements in immediately block; no nested blocks
description) ::=
<<
if ( !( <referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) ) {
    throw new org.antlr.runtime.tree.RewriteEarlyExitException();
}
while ( <referencedElements:{el | stream_<el>.hasNext()}; separator="||"> ) {
    <alt>
}
<referencedElements:{el | stream_<el>.reset();<\n>}>
>>

```

```

rewriteAlt(a) ::= <<
// <a.description>
<if(a.pred)>
if (<a.pred> ) {
    <a.alt>
}<\n>
<else>
{
    <a.alt>
}
}

```

```

}<\n>
<endif>
>>

/** For empty rewrites: "r : ... -> ;" */
rewriteEmptyAlt() ::= "root_0 = null;"

rewriteTree(root,children,description,enclosingTreeLevel,treeLevel) ::= <<
// <fileName>:<description>
{
var root_<treeLevel> = this.adaptor.nil();
<root:rewriteElement()>
<children:rewriteElement()>
this.adaptor.addChild(root_<enclosingTreeLevel>, root_<treeLevel>);
}<\n>
>>

rewriteElementList(elements) ::= "<elements:rewriteElement()>"

rewriteElement(e) ::= <<
<@pregen()>
<e.el>
>>

/** Gen ID or ID[args] */
rewriteTokenRef(token,elementIndex,terminalOptions,args) ::= <<
this.adaptor.addChild(root_<treeLevel>, <createRewriteNodeFromElement(...)>);<\n>
>>

/** Gen $label ... where defined via label=ID */
rewriteTokenLabelRef(label,elementIndex) ::= <<
this.adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());<\n>
>>

/** Gen $label ... where defined via label+=ID */
rewriteTokenListLabelRef(label,elementIndex) ::= <<
this.adaptor.addChild(root_<treeLevel>, stream_<label>.nextNode());<\n>
>>

/** Gen ^($label ...) */
rewriteTokenLabelRefRoot(label,elementIndex) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);<\n>
>>

/** Gen ^($label ...) where label+=... */
rewriteTokenListLabelRefRoot ::= rewriteTokenLabelRefRoot

/** Gen ^(ID ...) or ^(ID[args] ...) */

```

```

rewriteTokenRefRoot(token,elementIndex,terminalOptions,args) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(<createRewriteNodeFromElement(...)>, root_<treeLevel>);<\n>
>>

rewriteImaginaryTokenRef(args,token,terminalOptions,elementIndex) ::= <<
this.adaptor.addChild(root_<treeLevel>, <createImaginaryNode(tokenType=token, ...)>);<\n>
>>

rewriteImaginaryTokenRefRoot(args,token,terminalOptions,elementIndex) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(<createImaginaryNode(tokenType=token, ...)>,
root_<treeLevel>);<\n>
>>

/** plain -> {foo} action */
rewriteAction(action) ::= <<
root_0 = <action>;<\n>
>>

/** What is the name of the previous value of this rule's root tree? This
* let's us refer to $rule to mean previous value. I am reusing the
* variable 'tree' sitting in retval struct to hold the value of root_0 right
* before I set it during rewrites. The assign will be to retval.tree.
*/
prevRuleRootRef() ::= "retval"

rewriteRuleRef(rule) ::= <<
this.adaptor.addChild(root_<treeLevel>, stream_<rule>.nextTree());<\n>
>>

rewriteRuleRefRoot(rule) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(stream_<rule>.nextNode(), root_<treeLevel>);<\n>
>>

rewriteNodeAction(action) ::= <<
this.adaptor.addChild(root_<treeLevel>, <action>);<\n>
>>

rewriteNodeActionRoot(action) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(<action>, root_<treeLevel>);<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel=rule */
rewriteRuleLabelRef(label) ::= <<
this.adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

/** Gen $ruleLabel ... where defined via ruleLabel+=rule */
rewriteRuleListLabelRef(label) ::= <<

```

```

this.adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel=rule */
rewriteRuleLabelRefRoot(label) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);<\n>
>>

/** Gen ^($ruleLabel ...) where ruleLabel+=rule */
rewriteRuleListLabelRefRoot(label) ::= <<
root_<treeLevel> = this.adaptor.becomeRoot(stream_<label>.nextNode(), root_<treeLevel>);<\n>
>>

rewriteWildcardLabelRef(label) ::= <<
this.adaptor.addChild(root_<treeLevel>, stream_<label>.nextTree());<\n>
>>

createImaginaryNode(tokenType,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
<! new MethodNode(IDLabel, args) !>
new <terminalOptions.node>(<tokenType><if(args)>, <args; separator=","><endif>)
<else>
this.adaptor.create(<tokenType>, <args; separator=","><if(!args)>"<tokenType>"<endif>)
<endif>
>>

createRewriteNodeFromElement(token,terminalOptions,args) ::= <<
<if(terminalOptions.node)>
new <terminalOptions.node>(stream_<token>.nextToken(<if(args)>, <args; separator=","><endif>))
<else>
<if(args)> <! must create new node from old !>
this.adaptor.create(<token>, <args; separator=",">)
<else>
stream_<token>.nextNode()
<endif>
<endif>
>>

```

Found in path(s):

```

* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/JavaScript/AST.stg

```

No license file was found, but licenses were detected in source scan.

```

/*
* [The "BSD license"]
* Copyright (c) 2010 Terence Parr and Alan Condit
* Copyright (c) 2006 Kay Roepke (Objective-C runtime)
* All rights reserved.

```

```
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/
```

Found in path(s):

```
*/opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-jar/org/antlr/codegen/ObjCTarget.java
```

No license file was found, but licenses were detected in source scan.

```
/*
* [The "BSD license"]
* Copyright (c) 2007-2008 Johannes Luber
* Copyright (c) 2005-2007 Kunle Odutola
* Copyright (c) 2011 Sam Harwell
* Copyright (c) 2011 Terence Parr
* All rights reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in the
* documentation and/or other materials provided with the distribution.
* 3. The name of the author may not be used to endorse or promote products
* derived from this software without specific prior written permission.
*
* THIS SOFTWARE IS PROVIDED BY THE AUTHOR ``AS IS" AND ANY EXPRESS OR
```



```

* IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED.
* IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT,
* INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
* NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE,
* DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY
* THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT
* (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF
* THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
*/

```

```

csharpVisibilityMap ::= [
  "private":"private",
  "protected":"protected",
  "public":"public",
  "fragment":"private",
  default:"private"
]

```

```

/** The overall file structure of a recognizer; stores methods for rules
* and cyclic DFAs plus support code.
*/

```

```

outputFile( LEXER,PARSER,TREE_PARSER, actionScope, actions,
  docComment, recognizer,
  name, tokens, tokenNames, rules, cyclicDFAs,
  bitsets, buildTemplate, buildAST, rewriteMode, profile,
  backtracking, synpreds, memoize, numRules,
  fileName, ANTLRVersion, generatedTimestamp, trace,
  scopes, superClass, literals) ::=

```

```
<<
```

```

//-----
// \<auto-generated>
//   This code was generated by a tool.
//   ANTLR Version: <ANTLRVersion>
//
//   Changes to this file may cause incorrect behavior and will be lost if
//   the code is regenerated.
// \</auto-generated>
//-----

```

```
// $ANTLR <ANTLRVersion> <fileName> <generatedTimestamp>
```

```

<if(trace)>
#define ANTLR_TRACE
<endif>
<@debugPreprocessor()>
// The variable 'variable' is assigned but its value is never used.
#pragma warning disable 168, 219

```

```

// Unreachable code detected.
#pragma warning disable 162
// Missing XML comment for publicly visible type or member 'Type_or_Member'
#pragma warning disable 1591

<actions.(actionScope).header>

<@imports>
using System.Collections.Generic;
using Antlr.Runtime;
using Antlr.Runtime.Misc;
<if(TREE_PARSER)>
using Antlr.Runtime.Tree;
using RewriteRuleITokenStream = Antlr.Runtime.Tree.RewriteRuleTokenStream;
<endif>
using ConditionalAttribute = System.Diagnostics.ConditionalAttribute;
<@end>
<if(actions.(actionScope).namespace)>
namespace <actions.(actionScope).namespace>
{
<endif>
<docComment>
<recognizer>
<if(actions.(actionScope).namespace)>

} // namespace <actions.(actionScope).namespace>
<endif>
>>

lexerInputStreamType() ::= <<
<actions.(actionScope).inputStreamType; null="ICharStream">
>>

lexer(grammar, name, tokens, scopes, rules, numRules, filterMode, labelType="CommonToken",
superClass={ <if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Lexer
<endif>} ) ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<ANTLRVersion>")]
[System.CLSCompliant(false)]
<parserModifier(grammar=grammar, actions=actions)> partial class <grammar.recognizerName> :
<@superClassName><superClass><@end>
{
<tokens:{it|public const int <it.name; format="id">=<it.type>;}; separator="\n">
<scopes:{it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif>}>
<actions.lexer.members>

// delegates
<grammar.delegates:
{g|private <g.recognizerName> <g.delegateName()>;}; separator="\n">

```

```

// delegators
<grammar.delegators:
  {g|private <g.recognizerName> <g.delegateName(>);}; separator="\n">
<last(grammar.delegators):{g|private <g.recognizerName> gParent;}>

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<! needed by subclasses !>
{
  OnCreated();
}

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<lexerInputStreamType(>
input<grammar.delegators:{g|, <g.recognizerName> <g.delegateName(>}&> )
: this(input, new RecognizerSharedState(<grammar.delegators:{g|, <g.delegateName(>}&>))
{
}

<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<lexerInputStreamType(> input,
RecognizerSharedState state<grammar.delegators:{g|, <g.recognizerName> <g.delegateName(>}&>))
: base(input, state)
{
<if(memoize)>
<if(grammar.grammarIsRoot)>
  state.ruleMemo = new System.Collections.Generic.Dictionary<int, int>[<numRules>+1];<\n><! index from 1..n !>
<endif>
<endif>
<grammar.directDelegates:
  {g|<g.delegateName(> = new <g.recognizerName>(input, this.state<trunc(g.delegators):{p|,
<p.delegateName(>}&>, this);}; separator="\n">
<grammar.delegators:
  {g|this.<g.delegateName(> = <g.delegateName(>);}; separator="\n">
<last(grammar.delegators):{g|gParent = <g.delegateName(>};}>

  OnCreated();
}
public override string GrammarFileName { get { return "<fileName>"; } }

private static readonly bool[] decisionCanBacktrack = new bool[0];

<if(grammar.hasDelegates)>
public override <lexerInputStreamType(> CharStream
{
  get
  {
    return base.CharStream;
  }
  set
  {
    base.CharStream = value;

```

```

    <grammar.directDelegates:
      {g|<g:delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|, <p:delegateName()>}>,
this);}; separator="\n">
    <grammar.delegators:
      {g|this.<g:delegateName()> = <g:delegateName()>;}; separator="\n">
      <last(grammar.delegators):{g|gParent = <g:delegateName()>;}>
    }
  }

<if(grammar.delegates)>
public override void SetState(RecognizerSharedState state)
{
  base.SetState(state);
  <grammar.delegates:{g|<g:delegateName()>.SetState(state);}; separator="\n">
}
<endif>

<endif>
<if(filterMode)>
  <filteringNextToken()>
<endif>

[Conditional("ANTLR_TRACE")]
protected virtual void OnCreated() {}
[Conditional("ANTLR_TRACE")]
protected virtual void EnterRule(string ruleName, int ruleIndex) {}
[Conditional("ANTLR_TRACE")]
protected virtual void LeaveRule(string ruleName, int ruleIndex) {}

  <rules; separator="\n">

  <insertLexerSynpreds(synpreds)>

  #region DFA
  <cyclicDFAs:{ dfa | DFA<dfa.decisionNumber> dfa<dfa.decisionNumber>;}; separator="\n">

  protected override void InitDFAs()
  {
    base.InitDFAs();
    <cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = new DFA<dfa.decisionNumber>(this<if(dfa.specialStateSTs)>,
SpecialStateTransition<dfa.decisionNumber><endif>;);}; separator="\n">
  }

  <cyclicDFAs:cyclicDFA()> <! dump tables for all DFA !>
  #endregion

  }
  >>

```

```

/** A override of Lexer.nextToken() that backtracks over mTokens() looking
 * for matches. No error can be generated upon error; just rewind, consume
 * a token and then try again. backtracking needs to be set as well.
 * Make rule memoization happen only at levels above 1 as we start mTokens
 * at backtracking==1.
 */
filteringNextToken() ::= <<
public override IToken NextToken()
{
    while (true)
    {
        if (input.LA(1) == CharStreamConstants.EndOfFile)
        {
            IToken eof = new CommonToken((ICharStream)input, CharStreamConstants.EndOfFile, TokenChannels.Default,
input.Index, input.Index);
            eof.Line = Line;
            eof.CharPositionInLine = CharPositionInLine;
            return eof;
        }
        state.token = null;
        state.channel = TokenChannels.Default;
        state.tokenStartCharIndex = input.Index;
        state.tokenStartCharPositionInLine = input.CharPositionInLine;
        state.tokenStartLine = input.Line;
        state.text = null;
        try
        {
            int m = input.Mark();
            state.backtracking=1;<! means we won't throw slow exception !>
            state.failed=false;
            mTokens();
            state.backtracking=0;
            <! mTokens backtracks with synpred at backtracking==2
            and we set the synpredgate to allow actions at level 1. !>
            if (state.failed)
            {
                input.Rewind(m);
                input.Consume();<! advance one char and try again !>
            }
            else
            {
                Emit();
                return state.token;
            }
        }
        catch (RecognitionException re)
        {

```

```

    // shouldn't happen in backtracking mode, but...
    ReportError(re);
    Recover(re);
}
}
}

public override void Memoize(IIntStream input, int ruleIndex, int ruleStartIndex)
{
    if (state.backtracking > 1)
        base.Memoize(input, ruleIndex, ruleStartIndex);
}

public override bool AlreadyParsedRule(IIntStream input, int ruleIndex)
{
    if (state.backtracking > 1)
        return base.AlreadyParsedRule(input, ruleIndex);

    return false;
}
>>

actionGate() ::= "state.backtracking == 0"

filteringActionGate() ::= "state.backtracking == 1"

/** How to generate a parser */
genericParser(grammar, name, scopes, tokens, tokenNames, rules, numRules,
    bitsets, inputStreamType, superClass,
    labelType, members, rewriteElementType,
    filterMode, ASTLabelType="object") ::= <<
[System.CodeDom.Compiler.GeneratedCode("ANTLR", "<ANTLRVersion>")]
[System.CLSCompliant(false)]
<parserModifier(grammar=grammar, actions=actions)> partial class <grammar.recognizerName> :
<@superClassName><superClass><@end>
{
<if(grammar.grammarIsRoot)>
    internal static readonly string[] tokenNames = new string[] {
        "\<invalid>", "\<EOR>", "\<DOWN>", "\<UP>", <tokenNames; separator=", ">
    };
<endif>
    <tokens:{it|public const int <it.name; format="id">=<it.type>;}; separator="\n">

<if(grammar.delegates)>
    // delegates
    <grammar.delegates:
        {g|private <g.recognizerName> <g.delegateName(>);}; separator="\n">
<endif>

```

```

<if(grammar.delegators)>
// delegators
<grammar.delegators:
  {g|private <g.recognizerName> <g.delegateName(>); separator="\n">
<last(grammar.delegators):{g|private <g.recognizerName> gParent;}>
<endif>

<if(grammar.delegates)>
public override void SetState(RecognizerSharedState state)
{
  base.SetState(state);
  <grammar.delegates:{g|<g.delegateName(>.SetState(state);}; separator="\n">
}

<if(TREE_PARSER)>
public override void SetTreeNodeStream(ITreeNodeStream input)
{
  base.SetTreeNodeStream(input);
  <grammar.delegates:{g|<g.delegateName(>.SetTreeNodeStream(input);}; separator="\n">
}
<endif>
<endif>

<scopes: {it|<if(it.isDynamicGlobalScope)><globalAttributeScope(scope=it)><endif> }>
<@members(>

public override string[] TokenNames { get { return
<grammar.composite.rootGrammar.recognizerName>.tokenNames; } }
public override string GrammarFileName { get { return "<fileName>"; } }

<members>

[Conditional("ANTLR_TRACE")]
protected virtual void OnCreated() {}
[Conditional("ANTLR_TRACE")]
protected virtual void EnterRule(string ruleName, int ruleIndex) {}
[Conditional("ANTLR_TRACE")]
protected virtual void LeaveRule(string ruleName, int ruleIndex) {}

#region Rules
<rules; separator="\n">
#endregion Rules

<if(grammar.delegatedRules)>
<! generate rule/method definitions for imported rules so they
  appear to be defined in this recognizer. !>
#region Delegated rules
<grammar.delegatedRules:{ruleDescriptor|

```

```

<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> <returnType(ruleDescriptor)>
<ruleDescriptor.name; format="id"><ruleDescriptor.parameterScope:parameterScope()> <!throws
RecognitionException !>{ <if(ruleDescriptor.hasReturnValue)>return
<endif><ruleDescriptor.grammar:delegateName()>.<ruleDescriptor.name;
format="id"><ruleDescriptor.parameterScope><ruleDescriptor.parameterScope.attributes:{ a|<a.name;
format="id">}; separator=", "><endif>}; \}}; separator="\n">
#endregion Delegated rules
<endif>

<insertSynpreds(synpreds)>

<if(cyclicDFAs)>
#region DFA
<cyclicDFAs:{ dfa | private DFA<dfa.decisionNumber> dfa<dfa.decisionNumber>;}; separator="\n">

protected override void InitDFAs()
{
base.InitDFAs();
<cyclicDFAs:{ dfa | dfa<dfa.decisionNumber> = new DFA<dfa.decisionNumber>( this<if(dfa.specialStateSTs)>,
SpecialStateTransition<dfa.decisionNumber><endif> );}; separator="\n">
}

<cyclicDFAs:cyclicDFA()><! dump tables for all DFA !>
#endregion DFA
<endif>

<if(bitsets)>
#region Follow sets
private static class Follow
{
<bitsets:{ it|<bitset(name={ _<it.name>_in_<it.inName><it.tokenIndex>}, words64=it.bits)>}; separator="\n">
}
#endregion Follow sets
<endif>
}
>>

@genericParser.members() ::= <<
#if ANTLR_DEBUG
private static readonly bool[] decisionCanBacktrack =
new bool[]
{
false, // invalid decision
<grammar.decisions:{ d | <d.dfa.hasSynPred>}; wrap="\n", separator=", ">
};
#else
private static readonly bool[] decisionCanBacktrack = new bool[0];
#endif

```



```

<! WARNING. bug in ST: this is cut-n-paste into Dbg.stg !>
<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<inputStreamType>
input<grammar.delegators:{g|, <g.recognizerName> <g.delegateName()>}>)
: this(input, new RecognizerSharedState()<grammar.delegators:{g|, <g.delegateName()>}>)
{
}
<actions.(actionScope).ctorModifier; null="public"> <grammar.recognizerName>(<inputStreamType> input,
RecognizerSharedState state<grammar.delegators:{g|, <g.recognizerName> <g.delegateName()>}>)
: base(input, state)
{
<if(grammar.directDelegates)>
<grammar.directDelegates:
{g|<g.delegateName()> = new <g.recognizerName>(input, state<trunc(g.delegators):{p|, <p.delegateName()>}>,
this);}; separator="\n">
<endif>
<if(grammar.indirectDelegates)>
<grammar.indirectDelegates:{g | <g.delegateName()> = <g.delegator.delegateName()>.<g.delegateName()>};
separator="\n">
<endif>
<if(grammar.delegators)>
<last(grammar.delegators):{g|gParent = <g.delegateName()>};>
<endif>
<parserCtorBody()>
OnCreated();
}
>>

// imported grammars are 'public' (can't be internal because their return scope classes must be accessible)
parserModifier(grammar, actions) ::= <<
<if(grammar.grammarIsRoot)><actions.(actionScope).modifier; null="public"><else>public<endif>
>>

parserCtorBody() ::= <<
<if(memoize)>
<if(grammar.grammarIsRoot)>
this.state.ruleMemo = new System.Collections.Generic.Dictionary<int,
int>[<length(grammar.allImportedRules)>+1];<\n><! index from 1..n !>
<endif>
<endif>
<grammar.delegators:
{g|this.<g.delegateName()> = <g.delegateName()>}; separator="\n">
>>

parser(grammar, name, scopes, tokens, tokenNames, rules, numRules, bitsets,
ASTLabelType="object",
superClass={<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Parser
<endif>}, labelType="IToken",
members={<actions.parser.members>}) ::= <<

```

```

<genericParser(inputStreamType="ITokenStream", rewriteElementType="IToken", ...)>
>>

/** How to generate a tree parser; same as parser except the input
 * stream is a different type.
 */
treeParser(grammar, name, scopes, tokens, tokenNames, globalAction, rules,
    numRules, bitsets, filterMode, labelType={<ASTLabelType>}, ASTLabelType="object",
    superClass={<if(actions.(actionScope).superClass)><actions.(actionScope).superClass><else>Antlr.Runtime.Tree.<
if(filterMode)><if(buildAST)>TreeRewriter<else>TreeFilter<endif><else>TreeParser<endif><endif>},
    members={<actions.treeparser.members>}) ::= <<
<genericParser(inputStreamType="ITreeNodeStream", rewriteElementType="Node", ...)>
>>

/** A simpler version of a rule template that is specific to the imaginary
 * rules created for syntactic predicates. As they never have return values
 * nor parameters etc..., just give simplest possible method. Don't do
 * any of the normal memoization stuff in here either; it's a waste.
 * As predicates cannot be inlined into the invoking rule, they need to
 * be in a rule by themselves.
 */
synpredRule(ruleName, ruleDescriptor, block, description, nakedBlock) ::=
<<
[Conditional("ANTLR_TRACE")]
protected virtual void EnterRule_<ruleName>_fragment() {}
[Conditional("ANTLR_TRACE")]
protected virtual void LeaveRule_<ruleName>_fragment() {}

// $ANTLR start <ruleName>
<ruleModifier(grammar,ruleDescriptor)> void
<ruleName>_fragment(<ruleDescriptor.parameterScope:parameterScope()>)
{
    <ruleLabelDefs()>
    EnterRule_<ruleName>_fragment();
    EnterRule("<ruleName>_fragment", <ruleDescriptor.index>);
    TraceIn("<ruleName>_fragment", <ruleDescriptor.index>);
    try
    {
        <block>
    }
    finally
    {
        TraceOut("<ruleName>_fragment", <ruleDescriptor.index>);
        LeaveRule("<ruleName>_fragment", <ruleDescriptor.index>);
        LeaveRule_<ruleName>_fragment();
    }
}
// $ANTLR end <ruleName>

```

>>

```
insertLexerSynpreds(synpreds) ::= <<
<insertSynpreds(synpreds)>
>>
```

```
insertSynpreds(synpreds) ::= <<
<if(synpreds)>
#region Synpreds
private bool EvaluatePredicate(System.Action fragment)
{
    bool success = false;
    state.backtracking++;
    <@start(>
    try { DebugBeginBacktrack(state.backtracking);
    int start = input.Mark();
    try
    {
        fragment();
    }
    catch ( RecognitionException re )
    {
        System.Console.Error.WriteLine("impossible: "+re);
    }
    success = !state.failed;
    input.Rewind(start);
    } finally { DebugEndBacktrack(state.backtracking, success); }
    <@stop(>
    state.backtracking--;
    state.failed=false;
    return success;
}
#endregion Synpreds
<endif>
>>
```

```
ruleMemoization(name) ::= <<
<if(memoize)>
if (state.backtracking > 0 && AlreadyParsedRule(input, <ruleDescriptor.index>)) { <returnFromRule(> }
<endif>
>>
```

```
/** How to test for failure and return from rule */
checkRuleBacktrackFailure() ::= <<
<if(backtracking)>if (state.failed) <returnFromRule(><endif>
>>
```

```
/** This rule has failed, exit indicating failure during backtrack */
```

```

ruleBacktrackFailure() ::= <<
<if(backtracking)>if (state.backtracking>0) {state.failed=true; <returnFromRule(>><endif>
>>

ruleWrapperMap ::= [
"bottomup":{<ruleWrapperBottomup(>>},
"topdown":{<ruleWrapperTopdown(>>},
default:""
]

ruleWrapperBottomup() ::= <<
<if(TREE_PARSER && filterMode)>
protected override <if(buildAST)>IAstRuleReturnScope<else>void<endif> Bottomup() { return bottomup(); }
<endif>
>>

ruleWrapperTopdown() ::= <<
<if(TREE_PARSER && filterMode)>
protected override <if(buildAST)>IAstRuleReturnScope<else>void<endif> Topdown() { return topdown(); }
<endif>
>>

/** How to generate code for a rule. This includes any return type
 * data aggregates required for multiple return values.
 */
rule(ruleName,ruleDescriptor,block,emptyRule,description,exceptions,finally,memoize) ::= <<
<ruleAttributeScope(scope=ruleDescriptor.ruleScope)>
<returnScope(ruleDescriptor.returnScope)>

[Conditional("ANTLR_TRACE")]
protected virtual void EnterRule_<ruleName>() { }
[Conditional("ANTLR_TRACE")]
protected virtual void LeaveRule_<ruleName>() {}
<ruleWrapperMap.(ruleName)>
// $ANTLR start "<ruleName>"
// <fileName>:<description>
[GrammarRule("<ruleName>")]
<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> <returnType(ruleDescriptor)> <ruleName;
format="id">(<ruleDescriptor.parameterScope:parameterScope(>>)
{
EnterRule_<ruleName>();
EnterRule("<ruleName>", <ruleDescriptor.index>);
TraceIn("<ruleName>", <ruleDescriptor.index>);
<ruleScopeSetUp(>
<ruleDeclarations(>
<ruleLabelDefs(>
<ruleDescriptor.actions.init>
try { DebugEnterRule(GrammarFileName, "<ruleName>");

```

```

DebugLocation(<ruleDescriptor.tree.line>, <ruleDescriptor.EORNode.charPositionInLine>);
<@preamble()>
try
{
<ruleMemoization(name=ruleName)>
<block>
<ruleCleanUp()>
<(ruleDescriptor.actions.after):execAction()>
}
<if(exceptions)>
<exceptions:{e|<catch(decl=e.decl,action=e.action)><\n>}>
<else>
<if(!emptyRule)>
<if(actions.(actionScope).rulecatch)>
<actions.(actionScope).rulecatch>
<else>
catch (RecognitionException re)
{
ReportError(re);
Recover(input,re);
<@setErrorReturnValue()>
}
<endif>
<endif>
<endif>
finally
{
TraceOut("<ruleName>", <ruleDescriptor.index>);
LeaveRule("<ruleName>", <ruleDescriptor.index>);
LeaveRule_<ruleName>();
<memoize()>
<ruleScopeCleanUp()>
<finally>
}
DebugLocation(<ruleDescriptor.EORNode.line>, <ruleDescriptor.EORNode.charPositionInLine>);
} finally { DebugExitRule(GrammarFileName, "<ruleName>"); }
<@postamble()>
<returnFromRule()><\n>
}
// $ANTLR end "<ruleName>"
>>

// imported grammars need to have internal rules
ruleModifier(grammar,ruleDescriptor) ::= <<
<if(grammar.grammarIsRoot)><csharpVisibilityMap.(ruleDescriptor.modifier);
null="private"><else>internal<endif>
>>

```

```

// imported grammars need to have public return scopes
returnScopeModifier(grammar,ruleDescriptor) ::= <<
<if(grammar.grammarIsRoot)><csharpVisibilityMap.(ruleDescriptor.modifier);
null="private"><else>public<endif>
>>

catch(decl,action) ::= <<
catch (<e.decl>)
{
<e.action>
}
>>

ruleDeclarations() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<returnType(ruleDescriptor)> retval = new <returnType(ruleDescriptor)>();
retval.Start = (<labelType>)input.LT(1);
<elseif(ruleDescriptor.returnScope)>
<ruleDescriptor.returnScope.attributes: { a |
<a.type> <a.name; format="id"> = <if(a.initValue)><a.initValue><else><initValue(a.type)><endif>;
}>
<endif>
<if(memoize)>
int <ruleDescriptor.name>_startIndex = input.Index;
<endif>
>>

ruleScopeSetUp() ::= <<
<ruleDescriptor.useScopes: { it|<it>_stack.Push(new <it>_scope());<it>_scopeInit(<it>_stack.Peek());};
separator="\n">
<ruleDescriptor.ruleScope: { it|<it.name>_stack.Push(new
<it.name>_scope());<it.name>_scopeInit(<it.name>_stack.Peek());}; separator="\n">
>>

ruleScopeCleanUp() ::= <<
<ruleDescriptor.useScopes: { it|<it>_scopeAfter(<it>_stack.Peek());<it>_stack.Pop();}; separator="\n">
<ruleDescriptor.ruleScope: { it|<it.name>_scopeAfter(<it.name>_stack.Peek());<it.name>_stack.Pop();};
separator="\n">
>>

ruleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,ruleDescriptor.tokenListLabels,ruleDescriptor.wildcardTreeLabels,ruleDescriptor.wil
dcardTreeListLabels]
: { it|<labelType> <it.label.text> = default(<labelType>);}; separator="\n"
>
<ruleDescriptor.tokenListLabels
: { it|List<<labelType>> list_<it.label.text> = null;}; separator="\n"
>

```

```

<[ruleDescriptor.ruleListLabels,ruleDescriptor.wildcardTreeListLabels]
  :{it|List\<<ASTLabelType>> list_<it.label.text> = null;}; separator="\n"
>
<ruleDescriptor.ruleLabels:ruleLabelDef(); separator="\n">
<ruleDescriptor.ruleListLabels:ruleLabelDef(); separator="\n">
>>

lexerRuleLabelDefs() ::= <<
<[ruleDescriptor.tokenLabels,
  ruleDescriptor.tokenListLabels,
  ruleDescriptor.ruleLabels]
  :{it|<labelType> <it.label.text> = default(<labelType>);}; separator="\n"
>
<[ruleDescriptor.charListLabels,
  ruleDescriptor.charLabels]
  :{it|int <it.label.text> = 0;}; separator="\n"
>
<[ruleDescriptor.tokenListLabels,
  ruleDescriptor.ruleListLabels]
  :{it|List\<<labelType>> list_<it.label.text> = null;}; separator="\n"
>
<ruleDescriptor.charListLabels:{it|List\<int> list_<it.label.text> = null;}; separator="\n"
>
>>

returnFromRule() ::= <%
return
<if(!ruleDescriptor.isSynPred)>
<if(ruleDescriptor.hasReturnValue)>
<if(ruleDescriptor.hasSingleReturnValue)>
<! This comment is a hack to make sure the following
  single space appears in the output. !> <ruleDescriptor.singleValueReturnName>
<else>
<!!> retval
<endif>
<endif>
<endif>
<endif>
;
%>

ruleCleanup() ::= <<
<if(ruleDescriptor.hasMultipleReturnValues)>
<if(!TREE_PARSER)>
retval.Stop = (<labelType>)input.LT(-1);
<endif>
<endif>
>>

```

```

memoize() ::= <<
<if(memoize)>
<if(backtracking)>
if (state.backtracking > 0) { Memoize(input, <ruleDescriptor.index>, <ruleDescriptor.name>_StartIndex); }
<endif>
<endif>
>>

/** How to generate a rule in the lexer; naked blocks are used for
 * fragment rules.
 */
lexerRule(ruleName,nakedBlock,ruleDescriptor,block,memoize) ::= <<

[Conditional("ANTLR_TRACE")]
protected virtual void EnterRule_<ruleName>() {}
[Conditional("ANTLR_TRACE")]
protected virtual void LeaveRule_<ruleName>() {}

// $ANTLR start "<ruleName>"
[GrammarRule("<ruleName>")]
<ruleModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> void
m<ruleName>(<ruleDescriptor.parameterScope:parameterScope(>>)
{
EnterRule_<ruleName>();
EnterRule("<ruleName>", <ruleDescriptor.index>);
TraceIn("<ruleName>", <ruleDescriptor.index>);
<ruleScopeSetUp(>
<ruleDeclarations(>
try
{
<if(nakedBlock)>
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs(>
<ruleDescriptor.actions.init>
<block>
<else>
int _type = <ruleName>;
int _channel = DefaultTokenChannel;
<ruleMemoization(name=ruleName)>
<lexerRuleLabelDefs(>
<ruleDescriptor.actions.init>
<block>
<ruleCleanUp(>
state.type = _type;
state.channel = _channel;
<(ruleDescriptor.actions.after):execAction(>
<endif>
}

```



```

finally
{
    TraceOut("<ruleName>", <ruleDescriptor.index>);
    LeaveRule("<ruleName>", <ruleDescriptor.index>);
    LeaveRule_<ruleName>();
    <ruleScopeCleanUp()>
    <memoize()>
}
}
// $ANTLR end "<ruleName>"
>>

/** How to generate code for the implicitly-defined lexer grammar rule
 * that chooses between lexer rules.
 */
tokensRule(ruleName,nakedBlock,args,block,ruleDescriptor) ::= <<

public override void mTokens()
{
    <block><\n>
}
>>

// S U B R U L E S

/** A (...) subrule with multiple alternatives */
block(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
int alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
try { DebugEnterSubRule(<decisionNumber>);
try { DebugEnterDecision(<decisionNumber>, decisionCanBacktrack[<decisionNumber>]);
<decision>
} finally { DebugExitDecision(<decisionNumber>); }
<@postdecision()>
<@prebranch()>
switch (alt<decisionNumber>)
{
<alts:{a|<altSwitchCase(i,a)>}>
}
} finally { DebugExitSubRule(<decisionNumber>); }
<@postbranch()>
>>

/** A rule block with multiple alternatives */
ruleBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>

```

```

int alt<decisionNumber>=<maxAlt>;
<decls>
<@predecision()>
try { DebugEnterDecision(<decisionNumber>, decisionCanBacktrack[<decisionNumber>]);
<decision>
} finally { DebugExitDecision(<decisionNumber>); }
<@postdecision()>
switch (alt<decisionNumber>)
{
<alts:{a|altSwitchCase(i,a)}>
}
>>

```

```

ruleBlockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
DebugEnterAlt(1);
<alts>
<@postalt()>
>>

```

```

/** A special case of a (...) subrule with a single alternative */
blockSingleAlt(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,description) ::= <<
// <fileName>:<description>
<decls>
<@prealt()>
DebugEnterAlt(1);
<alts>
<@postalt()>
>>

```

```

/** A (..)+ block with 1 or more alternatives */
positiveClosureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::= <<
// <fileName>:<description>
int cnt<decisionNumber>=0;
<decls>
<@preloop()>
try { DebugEnterSubRule(<decisionNumber>);
while (true)
{
int alt<decisionNumber>=<maxAlt>;
<@predecision()>
try { DebugEnterDecision(<decisionNumber>, decisionCanBacktrack[<decisionNumber>]);
<decision>
} finally { DebugExitDecision(<decisionNumber>); }
<@postdecision()>

```

```

switch (alt<decisionNumber>)
{
<alts:{a|<altSwitchCase(i,a)>}>
default:
if (cnt<decisionNumber> >= 1)
goto loop<decisionNumber>;

<ruleBacktrackFailure()>
EarlyExitException eee<decisionNumber> = new EarlyExitException( <decisionNumber>, input );
DebugRecognitionException(eee<decisionNumber>);
<@earlyExitException()>
throw eee<decisionNumber>;
}
cnt<decisionNumber>++;
}
loop<decisionNumber>:
;

} finally { DebugExitSubRule(<decisionNumber>); }
<@postloop()>
>>

positiveClosureBlockSingleAlt ::= positiveClosureBlock

/** A (..)* block with 1 or more alternatives */
closureBlock(alts,decls,decision,enclosingBlockLevel,blockLevel,decisionNumber,maxK,maxAlt,description) ::=
<<
// <fileName>:<description>
<decls>
<@preloop()>
try { DebugEnterSubRule(<decisionNumber>);
while (true)
{
int alt<decisionNumber>=<maxAlt>;
<@predecision()>
try { DebugEnterDecision(<decisionNumber>, decisionCanBacktrack[<decisionNumber>]);
<decision>
} finally { DebugExitDecision(<decisionNumber>); }
<@postdecision()>
switch ( alt<decisionNumber> )
{
<alts:{a|<altSwitchCase(i,a)>}>
default:
goto loop<decisionNumber>;
}
}

loop<decisionNumber>:

```

```

;

} finally { DebugExitSubRule(<decisionNumber>); }
<@postloop()>
>>

closureBlockSingleAlt ::= closureBlock

/** Optional blocks (x)? are translated to (x|) by before code generation
 * so we can just use the normal block template
 */
optionalBlock ::= block

optionalBlockSingleAlt ::= block

/** A case in a switch that jumps to an alternative given the alternative
 * number. A DFA predicts the alternative and then a simple switch
 * does the jump to the code that actually matches that alternative.
 */
altSwitchCase(altNum,alt) ::= <<
case <altNum>:
<@prealt()>
DebugEnterAlt(<altNum>);
<alt>
break;<\n>
>>

/** An alternative is just a list of elements; at outermost level */
alt(elements,altNum,description,autoAST,outerAlt,treeLevel,rew) ::= <<
// <fileName>:<description>
{
<@declarations()>
<elements:element()>
<rew>
<@cleanup()>
}
>>

/** What to emit when there is no rewrite. For auto build
 * mode, does nothing.
 */
noRewrite(rewriteBlockLevel, treeLevel) ::= ""

// E L E M E N T S

/** Dump the elements one per line */
element(it) ::= <%
<@prematch()>

```

```

DebugLocation(<it.line>, <it.pos>);<\n>
<it.el><\n>
%>

/** match a token optionally with a label in front */
tokenRef(token,label,elementIndex,terminalOptions={}) ::= <<
<if(label)><label>=(<labelType>)<endif>Match(input,<token>,Follow._<token>_in_<ruleName><elementIndex>);
<checkRuleBacktrackFailure()>
>>

/** ids+=ID */
tokenRefAndListLabel(token,label,elementIndex,terminalOptions={}) ::= <<
<tokenRef(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

listLabelElem(label,elem,elemType) ::= <<
if (list_<label>==null) list_<label>=new List<<elemType; null={<labelType>}>>();
list_<label>.Add(<elem>);<\n>
>>

/** match a character */
charRef(char,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
Match(<char>); <checkRuleBacktrackFailure()>
>>

/** match a character range */
charRangeRef(a,b,label) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
MatchRange(<a>,<b>); <checkRuleBacktrackFailure()>
>>

/** For now, sets are interval tests and must be tested inline */
matchSet(s,label,elementIndex,postmatchCode="",terminalOptions={}) ::= <<
<if(label)>
<matchSetLabel()>
<endif>
if (<s>)
{
input.Consume();
<postmatchCode>
<if(!LEXER)>state.errorRecovery=false;<endif><if(backtracking)>state.failed=false;<endif>
}

```

```

else
{
<ruleBacktrackFailure()>
MismatchedSetException mse = new MismatchedSetException(null,input);
DebugRecognitionException(mse);
<@mismatchedSetException()>
<if(LEXER)>
Recover(mse);
throw mse;
<else>
throw mse;
<! use following code to make it recover inline; remove throw mse;
recoverFromMismatchedSet(input,mse,Follow._set_in_<ruleName><elementIndex>);
!>
<endif>
}<\n>
>>

matchSetUnchecked(s,label,elementIndex,postmatchCode=false) ::= <%
<if(label)>
<matchSetLabel()><\n>
<endif>
input.Consume();<\n>
<if(postmatchCode)>
<postmatchCode><\n>
<endif>
<if(!LEXER)>state.errorRecovery=false;<endif><if(backtracking)>state.failed=false;<endif>
%>

matchSetLabel() ::= <%
<if(LEXER)>
<label>= input.LA(1);
<else>
<label>=(<labelType>)input.LT(1);
<endif>
%>

matchRuleBlockSet ::= matchSet

matchSetAndListLabel(s,label,elementIndex,postmatchCode) ::= <<
<matchSet(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

/** Match a string literal */
lexerStringRef(string,label,elementIndex) ::= <%
<if(label)>
int <label>Start = CharIndex;<\n>

```

```

Match(<string>); <checkRuleBacktrackFailure()><\n>
int <label>StartLine<elementIndex> = Line;<\n>
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
<label> = new <labelType>(input, TokenTypes.Invalid, TokenChannels.Default, <label>Start, CharIndex-1);<\n>
<label>.Line = <label>StartLine<elementIndex>;<\n>
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
<else>
Match(<string>); <checkRuleBacktrackFailure()><\n>
<endif>
%>

```

```

wildcard(token,label,elementIndex,terminalOptions={ }) ::= <<
<if(label)>
<label>=(<labelType>)input.LT(1);<\n>
<endif>
MatchAny(input); <checkRuleBacktrackFailure()>
>>

```

```

wildcardAndListLabel(token,label,elementIndex,terminalOptions={ }) ::= <<
<wildcard(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

```

```

/** Match . wildcard in lexer */
wildcardChar(label, elementIndex) ::= <<
<if(label)>
<label> = input.LA(1);<\n>
<endif>
MatchAny(); <checkRuleBacktrackFailure()>
>>

```

```

wildcardCharListLabel(label, elementIndex) ::= <<
<wildcardChar(...)>
<listLabelElem(elem=label,elemType=labelType,...)>
>>

```

```

/** Match a rule reference by invoking it possibly with arguments
 * and a return value or values. The 'rule' argument was the
 * target rule name, but now is type Rule, whose toString is
 * same: the rule name. Now though you can access full rule
 * descriptor stuff.
 */
ruleRef(rule,label,elementIndex,args,scope) ::= <<
PushFollow(Follow._<rule.name>_in_<ruleName><elementIndex>);
<if(label)><label>=<endif><if(scope)><scope:delegateName()>.<endif><rule.name; format="id">(<args;
separator=", ">);
PopFollow();
<checkRuleBacktrackFailure()>

```

```
>>
```

```
/** ids+=r */
```

```
ruleRefAndListLabel(rule,label,elementIndex,args,scope) ::= <<
```

```
<ruleRef(...)>
```

```
<listLabelElem(elem=label,elemType={<ASTLabelType>},...)>
```

```
>>
```

```
/** A lexer rule reference.
```

```
*
```

```
* The 'rule' argument was the target rule name, but now
```

```
* is type Rule, whose toString is same: the rule name.
```

```
* Now though you can access full rule descriptor stuff.
```

```
*/
```

```
lexerRuleRef(rule,label,args,elementIndex,scope) ::= <%
```

```
<if(label)>
```

```
int <label>Start<elementIndex> = CharIndex;<\n>
```

```
int <label>StartLine<elementIndex> = Line;<\n>
```

```
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
```

```
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator="," ">);
```

```
<checkRuleBacktrackFailure()><\n>
```

```
<label> = new <labelType>(input, TokenTypes.Invalid, TokenChannels.Default, <label>Start<elementIndex>, CharIndex-1);<\n>
```

```
<label>.Line = <label>StartLine<elementIndex>;<\n>
```

```
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
```

```
<else>
```

```
<if(scope)><scope:delegateName().<endif>m<rule.name>(<args; separator="," ">);
```

```
<checkRuleBacktrackFailure()>
```

```
<endif>
```

```
%>
```

```
/** i+=INT in lexer */
```

```
lexerRuleRefAndListLabel(rule,label,args,elementIndex,scope) ::= <<
```

```
<lexerRuleRef(...)>
```

```
<listLabelElem(elem=label,elemType=labelType,...)>
```

```
>>
```

```
/** EOF in the lexer */
```

```
lexerMatchEOF(label,elementIndex) ::= <%
```

```
<if(label)>
```

```
int <label>Start<elementIndex> = CharIndex;<\n>
```

```
int <label>StartLine<elementIndex> = Line;<\n>
```

```
int <label>StartCharPos<elementIndex> = CharPositionInLine;<\n>
```

```
Match(EOF); <checkRuleBacktrackFailure()><\n>
```

```
<labelType> <label> = new <labelType>(input, EOF, TokenChannels.Default, <label>Start<elementIndex>, CharIndex-1);<\n>
```

```
<label>.Line = <label>StartLine<elementIndex>;<\n>
```

```
<label>.CharPositionInLine = <label>StartCharPos<elementIndex>;
```



```

<else>
Match(EOF); <checkRuleBacktrackFailure()>
<endif>
%>

// used for left-recursive rules
recRuleDefArg()          ::= "int <recRuleArg()>"
recRuleArg()             ::= "_p"
recRuleAltPredicate(ruleName,opPrec) ::= "<recRuleArg()> |<= <opPrec>"
recRuleSetResultAction() ::= "root_0=$<ruleName>_primary.tree;"
recRuleSetReturnAction(src,name)    ::= "$<name>=$<src>.<name>;"

/** match ^(root children) in tree parser */
tree(root, actionsAfterRoot, children, nullableChildList,
    enclosingTreeLevel, treeLevel) ::= <<
<root:element()>
<actionsAfterRoot:element()>
<if(nullableChildList)>
if (input.LA(1) == TokenType.Down)
{
    Match(input, TokenType.Down, null); <checkRuleBacktrackFailure()>
    <children:element()>
    Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
}
<else>
Match(input, TokenType.Down, null); <checkRuleBacktrackFailure()>
<children:element()>
Match(input, TokenType.Up, null); <checkRuleBacktrackFailure()>
<endif>
>>

/** Every predicate is used as a validating predicate (even when it is
 * also hoisted into a prediction expression).
 */
validateSemanticPredicate(pred,description) ::= <<
if (!(<evalPredicate(...)>))
{
    <ruleBacktrackFailure()>
    throw new FailedPredicateException(input, "<ruleName>", "<description>");
}
>>

// F i x e d D F A (if-then-else)

dfaState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<n>
<edges; separator="\nelse ">
<if(!isTrue.(last(edges).labelExpr) && (!last(edges).predicates))>

```

```

else
{
<if(eotPredictsAlt)>
alt<decisionNumber> = <eotPredictsAlt>;
<else>
<ruleBacktrackFailure()>
NoViableAltException nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>,
input, <k>);
DebugRecognitionException(nvae);
<@noViableAltException()>
throw nvae;
<endif>
}
<endif>
>>

```

```

/** Same as a normal DFA state except that we don't examine lookahead
 * for the bypass alternative. It delays error detection but this
 * is faster, smaller, and more what people expect. For (X)? people
 * expect "if ( LA(1)==X ) match(X);" and that's it.
 */
dfaOptionalBlockState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<\n>
<edges; separator="\nelse ">
>>

```

```

/** A DFA state that is actually the loopback decision of a closure
 * loop. If end-of-token (EOT) predicts any of the targets then it
 * should act like a default clause (i.e., no error can be generated).
 * This is used only in the lexer so that for ('a')* on the end of a rule
 * anything other than 'a' predicts exiting.
 */
dfaLoopbackState(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<
int LA<decisionNumber>_<k> = input.LA(<k>);<\n>
<edges; separator="\nelse "><\n>
<if(eotPredictsAlt)>
<if(!edges)>
alt<decisionNumber> = <eotPredictsAlt>;<! if no edges, don't gen ELSE !>
<else>
else
{
alt<decisionNumber> = <eotPredictsAlt>;
}<\n>
<endif>
<endif>
>>

```

```

/** An accept state indicates a unique alternative has been predicted */

```

```
dfaAcceptState(alt) ::= "alt<decisionNumber> = <alt>";
```

```
/** A simple edge with an expression. If the expression is satisfied,  
 * enter to the target state. To handle gated productions, we may  
 * have to evaluate some predicates for this edge.  
 */
```

```
dfaEdge(labelExpr, targetState, predicates) ::= <<  
if ((<labelExpr><if(predicates)> && (<predicates><endif>)  
{  
  <targetState>  
}  
>>
```

```
// F i x e d D F A (switch case)
```

```
/** A DFA state where a SWITCH may be generated. The code generator  
 * decides if this is possible: CodeGenerator.canGenerateSwitch().  
 */
```

```
dfaStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<  
switch (input.LA(<k>))  
{  
  <edges; separator="\n">  
  default:  
  <if(eotPredictsAlt)>  
    alt<decisionNumber>=<eotPredictsAlt>;  
    break;<\n>  
  <else>  
  {  
    <ruleBacktrackFailure()>  
    NoViableAltException nvae = new NoViableAltException("<description>", <decisionNumber>, <stateNumber>,  
input, <k>);  
    DebugRecognitionException(nvae);  
    <@noViableAltException()>  
    throw nvae;  
  }  
  <endif>  
}<\n>  
>>
```

```
dfaOptionalBlockStateSwitch(k,edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<  
switch (input.LA(<k>))  
{  
  <edges; separator="\n">  
}<\n>  
>>
```

```
dfaLoopbackStateSwitch(k, edges,eotPredictsAlt,description,stateNumber,semPredState) ::= <<  
switch (input.LA(<k>))
```

```

{
<edges; separator="\n">
<if(eotPredictsAlt)>
default:
alt<decisionNumber>=<eotPredictsAlt>;
break;<\n>
<endif>
}<\n>
>>

dfaEdgeSwitch(labels, targetState) ::= <<
<labels: {it|case <it>:}; separator="\n">
{
<targetState>
}
break;
>>

// C y c l i c D F A

/** The code to initiate execution of a cyclic DFA; this is used
 * in the rule to predict an alt just like the fixed DFA case.
 * The <name> attribute is inherited via the parser, lexer, ...
 */
dfaDecision(decisionNumber,description) ::= <<
try
{
alt<decisionNumber> = dfa<decisionNumber>.Predict(input);
}
catch (NoViableAltException nvae)
{
DebugRecognitionException(nvae);
throw;
}
>>

/* Dump DFA tables as run-length-encoded Strings of octal values.
 * Can't use hex as compiler translates them before compilation.
 * These strings are split into multiple, concatenated strings.
 * Java puts them back together at compile time thankfully.
 * Java cannot handle large static arrays, so we're stuck with this
 * encode/decode approach. See analysis and runtime DFA for
 * the encoding methods.
 */
cyclicDFA(dfa) ::= <<
private class DFA<dfa.decisionNumber> : DFA
{
private const string DFA<dfa.decisionNumber>_eotS =

```

```

"<dfa.javaCompressedEOT; wrap="\n\t\>";
private const string DFA<dfa.decisionNumber>_eofS =
"<dfa.javaCompressedEOF; wrap="\n\t\>";
private const string DFA<dfa.decisionNumber>_minS =
"<dfa.javaCompressedMin; wrap="\n\t\>";
private const string DFA<dfa.decisionNumber>_maxS =
"<dfa.javaCompressedMax; wrap="\n\t\>";
private const string DFA<dfa.decisionNumber>_acceptS =
"<dfa.javaCompressedAccept; wrap="\n\t\>";
private const string DFA<dfa.decisionNumber>_specialS =
"<dfa.javaCompressedSpecial; wrap="\n\t\>>";
private static readonly string[] DFA<dfa.decisionNumber>_transitionS =
{
<dfa.javaCompressedTransition: {s|<s; wrap="\n\>"; separator=",\n">
};

private static readonly short[] DFA<dfa.decisionNumber>_eot =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_eotS);
private static readonly short[] DFA<dfa.decisionNumber>_eof =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_eofS);
private static readonly char[] DFA<dfa.decisionNumber>_min =
DFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_minS);
private static readonly char[] DFA<dfa.decisionNumber>_max =
DFA.UnpackEncodedStringToUnsignedChars(DFA<dfa.decisionNumber>_maxS);
private static readonly short[] DFA<dfa.decisionNumber>_accept =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_acceptS);
private static readonly short[] DFA<dfa.decisionNumber>_special =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_specialS);
private static readonly short[][] DFA<dfa.decisionNumber>_transition;

static DFA<dfa.decisionNumber>()
{
int numStates = DFA<dfa.decisionNumber>_transitionS.Length;
DFA<dfa.decisionNumber>_transition = new short[numStates][];
for ( int i=0; i < numStates; i++ )
{
DFA<dfa.decisionNumber>_transition[i] =
DFA.UnpackEncodedString(DFA<dfa.decisionNumber>_transitionS[i]);
}
}

public DFA<dfa.decisionNumber>( BaseRecognizer recognizer<if(dfa.specialStateSTs)>,
SpecialStateTransitionHandler specialStateTransition<endif> )
<if(dfa.specialStateSTs)>
: base(specialStateTransition)
<endif>
{
this.recognizer = recognizer;

```

```

this.decisionNumber = <dfa.decisionNumber>;
this.eot = DFA<dfa.decisionNumber>_eot;
this.eof = DFA<dfa.decisionNumber>_eof;
this.min = DFA<dfa.decisionNumber>_min;
this.max = DFA<dfa.decisionNumber>_max;
this.accept = DFA<dfa.decisionNumber>_accept;
this.special = DFA<dfa.decisionNumber>_special;
this.transition = DFA<dfa.decisionNumber>_transition;
}

public override string Description { get { return "<dfa.description>"; } }

public override void Error(NoViableAltException nvae)
{
    DebugRecognitionException(nvae);
}
}<\n>
<if(dfa.specialStateSTs)>
private int SpecialStateTransition<dfa.decisionNumber>(DFA dfa, int s, IIntStream _input)<! throws
NoViableAltException!>
{
    <if(LEXER)>
    IIntStream input = _input;
    <endif>
    <if(PARSER)>
    ITokenStream input = (ITokenStream)_input;
    <endif>
    <if(TREE_PARSER)>
    ITreeNodeStream input = (ITreeNodeStream)_input;
    <endif>
    int _s = s;
    s = -1;
    <! pull these outside the switch cases to save space on locals !>
    int LA<dfa.decisionNumber>_1 = input.LA(1);
    int index<dfa.decisionNumber>_1 = input.Index;
    switch (_s)
    {
        <dfa.specialStateSTs: {state |case <i0>:<! compressed special state numbers 0..n-1 !>
        <state>}; separator="\n">

        default:
            break;
    }

    if (s >= 0)
        return s;

    <if(backtracking)>

```

```

if (state.backtracking > 0) {state.failed=true; return -1;}
<endif>
NoViableAltException nvae = new NoViableAltException(dfa.Description, <dfa.decisionNumber>, _s, input);
dfa.Error(nvae);
throw nvae;
}
<endif>
>>

/** A state in a cyclic DFA; it's a special state and part of a big switch on
 * state.
 */
cyclicDFAState(decisionNumber,stateNumber,edges,needErrorClause,semPredState) ::= <<
{
<if(semPredState)>
<! get next lookahead symbol to test edges, then rewind !>
input.Rewind();
<endif>
<edges; separator="\nelse ">
<if(semPredState)>
<! return input cursor to state before we rewound !>
input.Seek(index<decisionNumber>_1);
<endif>
break;
}
>>

/** Just like a fixed DFA edge, test the lookahead and indicate what
 * state to jump to next if successful.
 */
cyclicDFAEdge(labelExpr, targetStateNumber, edgeNumber, predicates) ::= <<
if ((<labelExpr><if(predicates)> && (<predicates><endif>)) {s = <targetStateNumber>;}<\n>
>>

/** An edge pointing at end-of-token; essentially matches any char;
 * always jump to the target.
 */
eotDFAEdge(targetStateNumber,edgeNumber, predicates) ::= <<
s = <targetStateNumber>;<\n>
>>

// D F A E X P R E S S I O N S

andPredicates(left,right) ::= "(<left>&&<right>)"

orPredicates(operands) ::= "(<operands; separator=\\|\\>)"

```

```

notPredicate(pred) ::= "!(<evalPredicate(...)>)"

evalPredicate(pred,description) ::= "<pred>"

evalSynPredicate(pred,description) ::= "EvaluatePredicate(<pred>_fragment)"

lookaheadTest(atom,k,atomAsInt) ::= "LA<decisionNumber>_<k>===<atom>"

/** Sometimes a lookahead test cannot assume that LA(k) is in a temp variable
 * somewhere. Must ask for the lookahead directly.
 */
isolatedLookaheadTest(atom,k,atomAsInt) ::= "input.LA(<k>)===<atom>"

lookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::= <%
(LA<decisionNumber>_<k><ge()><lower> && LA<decisionNumber>_<k><le()><upper>
%>

isolatedLookaheadRangeTest(lower,upper,k,rangeNumber,lowerAsInt,upperAsInt) ::=
"(input.LA(<k>)<ge()><lower> && input.LA(<k>)<le()><upper>)"

le() ::= "\<="
ge() ::= ">="

setTest(ranges) ::= <<
<ranges; separator="|">
>>

// A T T R I B U T E S

attributeScope(scope) ::= <<
<if(scope)>
<if(scope.attributes)>
protected sealed partial class <scope.name>_scope
{
<scope.attributes:{ it|public <it.decl>;}; separator="\n">
}
<if(scope.actions.scopeinit)>
protected void <scope.name>_scopeInit( <scope.name>_scope scope )
{
<scope.actions.scopeinit>
}
<else>
protected virtual void <scope.name>_scopeInit( <scope.name>_scope scope ) { }
<endif>
<if(scope.actions.scopeafter)>
protected void <scope.name>_scopeAfter( <scope.name>_scope scope )
{
<scope.actions.scopeafter>
}
}

```



```

}
<else>
protected virtual void <scope.name>_scopeAfter( <scope.name>_scope scope ) {}
<endif>
protected readonly ListStack\<<scope.name>_scope> <scope.name>_stack = new
ListStack\<<scope.name>_scope>();
<endif>
<endif>
>>

globalAttributeScope(scope) ::= <<
<attributeScope(...)>
>>

ruleAttributeScope(scope) ::= <<
<attributeScope(...)>
>>

returnStructName(it) ::= "<it.name>_return"

returnType(ruleDescriptor) ::= <%
<if(ruleDescriptor.returnScope.attributes && ruleDescriptor.hasMultipleReturnValues)>
<ruleDescriptor.grammar.recognizerName>.<ruleDescriptor:returnStructName()>
<elseif(ruleDescriptor.hasMultipleReturnValues)>
<ruleReturnBaseType()>
<elseif(ruleDescriptor.hasSingleReturnValue)>
<ruleDescriptor.singleValueReturnType>
<else>
void
<endif>
%>

/** Generate the C# type associated with a single or multiple return
 * values.
 */
ruleLabelType(referencedRule) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
<ruleReturnBaseType()>
<elseif(referencedRule.hasSingleReturnValue)>
<referencedRule.singleValueReturnType>
<else>
void
<endif>
%>

delegateName(it) ::= <<
<if(it.label)><it.label><else>g<it.name><endif>
>>

```

```

/** Using a type to init value map, try to init a type; if not in table
 * must be an object, default value is "null".
 */
initValue(typeName) ::= <<
default(<typeName>)
>>

/** Define a rule label including default value */
ruleLabelDef(label) ::= <%
<ruleLabelType(referencedRule=label.referencedRule)> <label.label.text> =
<initValue(typeName=ruleLabelType(referencedRule=label.referencedRule))>;
%>

/** Define a return struct for a rule if the code needs to access its
 * start/stop tokens, tree stuff, attributes, ... Leave a hole for
 * subgroups to stick in members.
 */
returnScope(scope) ::= <<
<if(scope.attributes && ruleDescriptor.hasMultipleReturnValues)>
<returnScopeModifier(grammar=grammar,ruleDescriptor=ruleDescriptor)> sealed partial class
<ruleDescriptor:returnStructName()> : <ruleReturnBaseType()><@ruleReturnInterfaces()>
{
<scope.attributes: {it|public <it.decl>;}; separator="\n">
<@ruleReturnMembers()>
}
<endif>
>>

ruleReturnBaseType() ::= <%
<if(TREE_PARSER)>Tree<else>Parser<endif>RuleReturnScope\<<labelType>>
%>

@returnScope.ruleReturnMembers() ::= <<
>>

parameterScope(scope) ::= <<
<scope.attributes: {it|<it.decl>;}; separator=", ">
>>

parameterAttributeRef(attr) ::= <<
<attr.name; format="id">
>>

parameterSetAttributeRef(attr,expr) ::= <<
<attr.name; format="id"> =<expr>;
>>

```

```

scopeAttributeRef(scope,attr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack[<scope>_stack.Count - <negIndex> - 1].<attr.name; format="id">
<else>
<if(index)>
<scope>_stack[<index>].<attr.name; format="id">
<else>
<scope>_stack.Peek().<attr.name; format="id">
<endif>
<endif>
%>

```

```

scopeSetAttributeRef(scope,attr,expr,index,negIndex) ::= <%
<if(negIndex)>
<scope>_stack[<scope>_stack.Count - <negIndex> - 1].<attr.name; format="id"> = <expr>;
<else>
<if(index)>
<scope>_stack[<index>].<attr.name; format="id"> = <expr>;
<else>
<scope>_stack.Peek().<attr.name; format="id"> = <expr>;
<endif>
<endif>
%>

```

```

/** $x is either global scope or x is rule with dynamic scope; refers
* to stack itself not top of stack. This is useful for predicates
* like {$function.Count>0 && $function::name.Equals("foo")}?
*/

```

```

isolatedDynamicScopeRef(scope) ::= "<scope>_stack"

```

```

/** reference an attribute of rule; might only have single return value */
ruleLabelRef(referencedRule,scope,attr) ::= <%
<if(referencedRule.hasMultipleReturnValues)>
(<scope>!=null?(<returnType(referencedRule)><scope>).<attr.name; format="id">:<initValue(attr.type)>)
<else>
<scope>
<endif>
%>

```

```

returnAttributeRef(ruleDescriptor,attr) ::= <%
<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name; format="id">
<else>
<attr.name; format="id">
<endif>
%>

```

```

returnSetAttributeRef(ruleDescriptor,attr,expr) ::= <%

```

```

<if(ruleDescriptor.hasMultipleReturnValues)>
retval.<attr.name; format="id"> =<expr>;
<else>
<attr.name; format="id"> =<expr>;
<endif>
%>

/** How to translate $tokenLabel */
tokenLabelRef(label) ::= "<label>"

/** ids+=ID {$ids} or e+=expr {$e} */
listLabelRef(label) ::= "list_<label>"

// not sure the next are the right approach

tokenLabelPropertyRef_text(scope,attr) ::= "(<scope>!=null?<scope>.Text:default(string))"
tokenLabelPropertyRef_type(scope,attr) ::= "(<scope>!=null?<scope>.Type:0)"
tokenLabelPropertyRef_line(scope,attr) ::= "(<scope>!=null?<scope>.Line:0)"
tokenLabelPropertyRef_pos(scope,attr) ::= "(<scope>!=null?<scope>.CharPositionInLine:0)"
tokenLabelPropertyRef_channel(scope,attr) ::= "(<scope>!=null?<scope>.Channel:0)"
tokenLabelPropertyRef_index(scope,attr) ::= "(<scope>!=null?<scope>.TokenIndex:0)"
tokenLabelPropertyRef_tree(scope,attr) ::= "<scope>_tree"
tokenLabelPropertyRef_int(scope,attr) ::= "(<scope>!=null?int.Parse(<scope>.Text):0)"

ruleLabelPropertyRef_start(scope,attr) ::= "(<scope>!=null?((<labelType><scope>.Start):default(<labelType>))"
ruleLabelPropertyRef_stop(scope,attr) ::= "(<scope>!=null?((<labelType><scope>.Stop):default(<labelType>))"
ruleLabelPropertyRef_tree(scope,attr) ::=
"(<scope>!=null?((<ASTLabelType><scope>.Tree):default(<ASTLabelType>))"
ruleLabelPropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
(<scope>!=null?(input.TokenStream.ToString(
input.TreeAdaptor.GetTokenStartIndex(<scope>.Start),
input.TreeAdaptor.GetTokenStopIndex(<scope>.Start))):default(string))
<else>
(<scope>!=null?input.ToString(<scope>.Start,<scope>.Stop):default(string))
<endif>
%>

ruleLabelPropertyRef_st(scope,attr) ::= "(<scope>!=null?<scope>.Template:null)"

/** Isolated $RULE ref ok in lexer as it's a Token */
lexerRuleLabel(label) ::= "<label>"

lexerRuleLabelPropertyRef_type(scope,attr) ::=
"(<scope>!=null?<scope>.Type:0)"

lexerRuleLabelPropertyRef_line(scope,attr) ::=

```

```

"(<scope>!=null?<scope>.Line:0)"

lexerRuleLabelPropertyRef_pos(scope,attr) ::=
"(<scope>!=null?<scope>.CharPositionInLine:-1)"

lexerRuleLabelPropertyRef_channel(scope,attr) ::=
"(<scope>!=null?<scope>.Channel:0)"

lexerRuleLabelPropertyRef_index(scope,attr) ::=
"(<scope>!=null?<scope>.TokenIndex:0)"

lexerRuleLabelPropertyRef_text(scope,attr) ::=
"(<scope>!=null?<scope>.Text:default(string))"

lexerRuleLabelPropertyRef_int(scope,attr) ::=
"(<scope>!=null?int.Parse(<scope>.Text):0)"

// Somebody may ref $template or $tree or $stop within a rule:
rulePropertyRef_start(scope,attr) ::= "retval.Start"
rulePropertyRef_stop(scope,attr) ::= "retval.Stop"
rulePropertyRef_tree(scope,attr) ::= "retval.Tree"
rulePropertyRef_text(scope,attr) ::= <%
<if(TREE_PARSER)>
input.TokenStream.ToString(
input.TreeAdaptor.GetTokenStartIndex(retval.Start),
input.TreeAdaptor.GetTokenStopIndex(retval.Start))
<else>
input.ToString(retval.Start,input.LT(-1))
<endif>
%>
rulePropertyRef_st(scope,attr) ::= "retval.Template"

lexerRulePropertyRef_text(scope,attr) ::= "Text"
lexerRulePropertyRef_type(scope,attr) ::= "_type"
lexerRulePropertyRef_line(scope,attr) ::= "state.tokenStartLine"
lexerRulePropertyRef_pos(scope,attr) ::= "state.tokenStartCharPositionInLine"
lexerRulePropertyRef_index(scope,attr) ::= "-1" // undefined token index in lexer
lexerRulePropertyRef_channel(scope,attr) ::= "_channel"
lexerRulePropertyRef_start(scope,attr) ::= "state.tokenStartCharIndex"
lexerRulePropertyRef_stop(scope,attr) ::= "(CharIndex-1)"
lexerRulePropertyRef_int(scope,attr) ::= "int.Parse(<scope>.Text)"

// setting $st and $tree is allowed in local rule. everything else
// is flagged as error
ruleSetPropertyRef_tree(scope,attr,expr) ::= "retval.Tree = <expr>;"
ruleSetPropertyRef_st(scope,attr,expr) ::= "retval.Template =<expr>;"

/** How to execute an action (only when not backtracking) */

```

```

execAction(action) ::= <%
<if(backtracking)>
if (<actions.(actionScope).synpredgate><\n>
{<\n>
<@indentedAction()><\n>
}
<else>
<action>
<endif>
%>

@execAction.indentedAction() ::= <<
<action>
>>

/** How to always execute an action even when backtracking */
execForcedAction(action) ::= "<action>"

// M I S C (properties, etc...)

bitset(name, words64) ::= <<
public static readonly BitSet <name> = new BitSet(new ulong[] { <words64: {it|<it>UL}; separator="," > });
>>

codeFileExtension() ::= ".cs"

true_value() ::= "true"
false_value() ::= "false"

isTrue ::= [
"true" : true,
default : false
]

Found in path(s):
* /opt/cola/permits/1274703855_1645234694.1/0/antlr-3-5-2-sources-
jar/org/antlr/codegen/templates/CSharp2/CSharp2.stg

```

1.323 java-semver 0.9.0

1.323.1 Available under license :

No license file was found, but licenses were detected in source scan.

```

/*
* The MIT License
*
* Copyright 2012-2015 Zafar Khaja <zafarkhaja@gmail.com>.

```

*
 * Permission is hereby granted, free of charge, to any person obtaining a copy
 * of this software and associated documentation files (the "Software"), to deal
 * in the Software without restriction, including without limitation the rights
 * to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
 * copies of the Software, and to permit persons to whom the Software is
 * furnished to do so, subject to the following conditions:
 *
 * The above copyright notice and this permission notice shall be included in
 * all copies or substantial portions of the Software.
 *
 * THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
 * IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
 * FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
 * AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
 * LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
 * OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN
 * THE SOFTWARE.
 */

Found in path(s):

* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/UnexpectedCharacterException.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/LexerException.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/package-info.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/Greater.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/Expression.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/MetadataVersion.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/VersionParser.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/NormalVersion.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/Not.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/expr/ExpressionParser.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/package-info.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/util/package-info.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
 jar/com/github/zafarkhaja/semver/util/Stream.java
 * /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-

```

jar/com/github/zafarkhaja/semver/expr/UnexpectedTokenException.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/Less.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/LessOrEqual.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/Equal.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/NotEqual.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/GreaterOrEqual.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/Lexer.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/CompositeExpression.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/ParseException.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/And.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/expr/Or.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/Parser.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/Version.java
* /opt/cola/permits/1274705241_1645234225.55/0/java-semver-0-9-0-sources-
jar/com/github/zafarkhaja/semver/util/UnexpectedElementException.java

```

1.324 spock-framework---bill-of-materials 2.0-groovy-3.0

1.324.1 Available under license :

```

=====
== NOTICE file corresponding to the section 4 d of           ==
== the Apache License, Version 2.0,                          ==
== in this case for the Spock distribution.                   ==
=====

```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

It includes the following other software:

gentyref (<http://code.google.com/p/gentyref/>)

For licenses see the LICENSE file.

If any software distributed with Spock does not have an Apache 2 License, its license is explicitly listed in the LICENSE file.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain

separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the

origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.325 jersey-media-jaxb 2.32

1.325.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For
more information regarding authorship of content, please consult the listed
source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms
of the Eclipse Public License v. 2.0 which is available at
<http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made
available under the following Secondary Licenses when the conditions for such
availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU
General Public License, version 2 with the GNU Classpath Exception which is
available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in org.glassfish.jersey.server.validation.internal.hibernate

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>
- # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content
Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License,

Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was

received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and

to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is

to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and

can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the

executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other

circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW.

EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or

based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.326 micronaut-mqtt 2.1.0

1.326.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/annotation/Retained.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/bind/AnnotatedMqttBinder.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/bind/DefaultMqttBinderRegistry.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/serdes/MqttPayloadSerDesRegistry.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/bind/MqttBinderRegistry.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/bind/impl/QosMqttBinder.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/serdes/MqttPayloadSerDes.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/serdes/JavaLangMqttPayloadSerDes.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/config/MqttSSLConfiguration.java
* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-
jar/io/micronaut/mqtt/exception/MqttSubscriberException.java
```

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/RetainedMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/MqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/exception/MqttClientException.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/IdMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/TypedMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/serdes/JsonMqttPayloadSerDes.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/PayloadMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/MqttSubscriber.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/Qos.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/executor/MqttExecutorFactory.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/package-info.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/exception/DefaultMqttSubscriberExceptionHandler.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/intercept/MqttPublisherState.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/TopicMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/FallbackMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/Id.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/serdes/DefaultMqttPayloadSerDesRegistry.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/serdes/package-info.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/intercept/AbstractMqttIntroductionAdvice.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/Topics.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/AcknowledgeMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/intercept/AbstractMqttSubscriberAdvice.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/impl/DefaultMqttBinder.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/exception/MqttSubscriberExceptionHandler.java

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/bind/MqttBindingContext.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2021 original authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/cola/permits/1331473884_1653514363.4906576/0/micronaut-mqtt-core-2-1-0-sources-jar/io/micronaut/mqtt/annotation/Topic.java

1.327 jackson-integration-for-metrics 4.1.17

1.327.1 Available under license :

No license file was found, but licenses were detected in source scan.

Manifest-Version: 1.0

Automatic-Module-Name: com.codahale.metrics.json

Build-Jdk-Spec: 1.8

Bundle-Description: A set of Jackson modules which provide serializers for most Metrics classes.

Bundle-License: <https://www.apache.org/licenses/LICENSE-2.0.html>

Bundle-ManifestVersion: 2

Bundle-Name: Jackson Integration for Metrics

Bundle-SymbolicName: io.dropwizard.metrics.json

Bundle-Version: 4.1.17

Created-By: Apache Maven Bundle Plugin

Export-Package: com.codahale.metrics.json;uses:="com.codahale.metrics, com.fasterxml.jackson.core,com.fasterxml.jackson.databind";version="4.1.17"

Implementation-Title: Jackson Integration for Metrics

Implementation-Version: 4.1.17

Import-Package: com.codahale.metrics;version="[4.1,5)",com.codahale.metrics.health;resolution:=optional;version="[4.1,5)",com.fasterxml.jac

kson.core;version="[2.9,3)",com.fasterxml.jackson.databind;version="[2.9,3)",com.fasterxml.jackson.databind.module;version="[2.9,3)",com.fasterxml.jackson.databind.ser;version="[2.9,3)",com.fasterxml.jackson.databind.ser.std;version="[2.9,3)"

Private-Package: com.codahale.metrics.json

Require-Capability: osgi.ee;filter="(&(osgi.ee=JavaSE)(version=1.8))"

Tool: Bnd-5.1.1.202006162103

Found in path(s):

* /opt/cola/permits/1340188685_1654736889.2621648/0/metrics-json-4-1-17-jar/META-INF/MANIFEST.MF

1.328 kafka-protobuf-types 6.2.0

1.328.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
<modelVersion>4.0.0</modelVersion>
```

```
<parent>
```

```
<groupId>io.confluent</groupId>
```

```
<artifactId>kafka-schema-registry-parent</artifactId>
```

```
<version>6.2.0</version>
```

```
</parent>
```

```
<licenses>
```

```
<license>
```

```
<name>Confluent Community License</name>
```

```
<url>http://www.confluent.io/confluent-community-license</url>
```

```
<distribution>repo</distribution>
```

```
</license>
```

```
<license>
```

```
<name>Apache License 2.0</name>
```

```
<url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
```

```
<distribution>repo</distribution>
```

```
</license>
```

```
</licenses>
```

```
<artifactId>kafka-protobuf-types</artifactId>
```

```
<packaging>jar</packaging>
```

```
<name>kafka-protobuf-types</name>
```

```
<dependencies>
```

```
<dependency>
```

```
<groupId>com.google.protobuf</groupId>
```



```

    <artifactId>protobuf-java</artifactId>
  </dependency>
</dependency>
  <groupId>com.google.protobuf</groupId>
  <artifactId>protobuf-java-util</artifactId>
</dependency>
</dependency>
  <groupId>com.google.api.grpc</groupId>
  <artifactId>googleapis-common-protos</artifactId>
</dependency>
</dependency>
  <groupId>org.mockito</groupId>
  <artifactId>mockito-core</artifactId>
  <scope>test</scope>
</dependency>
</dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <groupId>com.github.os72</groupId>
      <artifactId>protoc-jar-maven-plugin</artifactId>
      <executions>
        <execution>
          <phase>generate-sources</phase>
          <goals>
            <goal>run</goal>
          </goals>
          <configuration>
            <protocVersion>${protobuf.version}</protocVersion>
            <includeStdTypes>true</includeStdTypes>
            <addSources>none</addSources>
            <addProtoSources>inputs</addProtoSources>
            <outputDirectory>src/main/java</outputDirectory>
            <inputDirectories>
              <include>src/main/proto</include>
            </inputDirectories>
          </configuration>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

</project>

Found in path(s):

* /opt/cola/permits/1340188433_1654734984.398556/0/kafka-protobuf-types-6-2-0-jar/META-INF/maven/io.confluent/kafka-protobuf-types/pom.xml

1.329 kotlinox-coroutines-core-common 1.3.2-1.3.60

1.329.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.330 fabric8-:::-kubernetes-model-:::- admission-registration,-authentication-and- authorization 4.10.3

1.330.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.331 kafka-schema-registry-client 5.5.1

1.331.1 Available under license :

Confluent Community License Agreement
Version 1.0

This Confluent Community License Agreement Version 1.0 (the Agreement) sets forth the terms on which Confluent, Inc. (Confluent) makes available certain software made available by Confluent under this Agreement (the Software). BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT. IF YOU DO NOT AGREE TO SUCH TERMS AND CONDITIONS, YOU MUST NOT USE THE SOFTWARE. IF YOU ARE RECEIVING THE SOFTWARE ON BEHALF OF A LEGAL ENTITY, YOU REPRESENT AND WARRANT THAT YOU HAVE THE ACTUAL AUTHORITY TO AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT ON BEHALF OF SUCH ENTITY. Licensee means you, an individual, or the entity on whose behalf you are receiving the Software.

1. LICENSE GRANT AND CONDITIONS.

1.1 License. Subject to the terms and conditions of this Agreement, Confluent hereby grants to Licensee a non-exclusive, royalty-free, worldwide, non-transferable, non-sublicenseable license during the term of this Agreement to: (a) use the Software; (b) prepare modifications and derivative works of the Software; (c) distribute the Software (including without limitation in source code or object code form); and (d) reproduce copies of the Software (the License). Licensee is not granted the right to, and Licensee shall not, exercise the License for an Excluded Purpose. For purposes of this Agreement, Excluded Purpose means making available any software-as-a-service, platform-as-a-service, infrastructure-as-a-service or other similar online service that competes with Confluent products or services that provide the Software.

1.2 Conditions. In consideration of the License, Licensees distribution of the Software is subject to the following conditions:

(a) Licensee must cause any Software modified by Licensee to carry prominent notices stating that Licensee modified the Software.

(b) On each Software copy, Licensee shall reproduce and not remove or alter all Confluent or third party copyright or other proprietary notices contained in the Software, and Licensee must provide the notice below with each copy.

This software is made available by Confluent, Inc., under the terms of the Confluent Community License Agreement, Version 1.0 located at <http://www.confluent.io/confluent-community-license>. BY INSTALLING, DOWNLOADING, ACCESSING, USING OR DISTRIBUTING ANY OF THE SOFTWARE, YOU AGREE TO THE TERMS OF SUCH LICENSE AGREEMENT.

1.3 Licensee Modifications. Licensee may add its own copyright notices to modifications made by Licensee and may provide additional or different license terms and conditions for use, reproduction, or distribution of Licensees modifications. While redistributing the Software or modifications thereof, Licensee may choose to offer, for a fee or free of

charge, support, warranty, indemnity, or other obligations. Licensee, and not Confluent, will be responsible for any such obligations.

1.4 No Sublicensing. The License does not include the right to sublicense the Software, however, each recipient to which Licensee provides the Software may exercise the Licenses so long as such recipient agrees to the terms and conditions of this Agreement.

2. TERM AND TERMINATION. This Agreement will continue unless and until earlier terminated as set forth herein. If Licensee breaches any of its conditions or obligations under this Agreement, this Agreement will terminate automatically and the License will terminate automatically and permanently.

3. INTELLECTUAL PROPERTY. As between the parties, Confluent will retain all right, title, and interest in the Software, and all intellectual property rights therein. Confluent hereby reserves all rights not expressly granted to Licensee in this Agreement. Confluent hereby reserves all rights in its trademarks and service marks, and no licenses therein are granted in this Agreement.

4. DISCLAIMER. CONFLUENT HEREBY DISCLAIMS ANY AND ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE.

5. LIMITATION OF LIABILITY. CONFLUENT WILL NOT BE LIABLE FOR ANY DAMAGES OF ANY KIND, INCLUDING BUT NOT LIMITED TO, LOST PROFITS OR ANY CONSEQUENTIAL, SPECIAL, INCIDENTAL, INDIRECT, OR DIRECT DAMAGES, HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, ARISING OUT OF THIS AGREEMENT. THE FOREGOING SHALL APPLY TO THE EXTENT PERMITTED BY APPLICABLE LAW.

6. GENERAL.

6.1 Governing Law. This Agreement will be governed by and interpreted in accordance with the laws of the state of California, without reference to its conflict of laws principles. If Licensee is located within the United States, all disputes arising out of this Agreement are subject to the exclusive jurisdiction of courts located in Santa Clara County, California, USA. If Licensee is located outside of the United States, any dispute, controversy or claim arising out of or relating to this Agreement will be referred to and finally determined by arbitration in accordance with the JAMS International Arbitration Rules. The tribunal will consist of one arbitrator. The place of arbitration will be Palo Alto, California. The language to be used in the arbitral proceedings will be English. Judgment upon the award rendered by the arbitrator may be entered in any court having jurisdiction thereof.

6.2 Assignment. Licensee is not authorized to assign its rights under this Agreement to any third party. Confluent may freely assign its rights under this Agreement to any third party.

6.3 Other. This Agreement is the entire agreement between the parties regarding the subject matter hereof. No amendment or modification of this Agreement will be valid or binding upon the parties unless made in writing and signed by the duly authorized representatives of both parties. In the event that any provision, including without limitation any condition, of this Agreement is held to be unenforceable, this Agreement and all licenses and rights granted hereunder will immediately terminate. Waiver by Confluent of a breach of any provision of this Agreement or the failure by Confluent to exercise any right hereunder will not be construed as a waiver of any subsequent breach of that right or as a waiver of any other right.

Apache Kafka

Copyright 2016 The Apache Software Foundation.

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

This distribution has a binary dependency on jersey, which is available under the CDDL License. The source code of jersey can be found at <https://github.com/jersey/jersey/>.
Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library. It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007. It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses. To find the details that apply to this artifact see the accompanying LICENSE file. For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses. Java ClassMate library was originally written by Tatu Saloranta (tatu.saloranta@iki.fi)

Other developers who have contributed code are:

* Brian Langel

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License"). See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta name="description" content="Home page of The Apache Software Foundation">

  <link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
  <link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
  <link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
  <link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
  <link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
  <link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
  <link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
  <link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
  <link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png">
  <link rel="icon" type="image/png" href="/favicons/favicon-32x32.png" sizes="32x32">
  <link rel="icon" type="image/png" href="/favicons/favicon-194x194.png" sizes="194x194">
  <link rel="icon" type="image/png" href="/favicons/favicon-96x96.png" sizes="96x96">
  <link rel="icon" type="image/png" href="/favicons/android-chrome-192x192.png" sizes="192x192">
  <link rel="icon" type="image/png" href="/favicons/favicon-16x16.png" sizes="16x16">
  <link rel="manifest" href="/favicons/manifest.json">
  <link rel="shortcut icon" href="/favicons/favicon.ico">
  <meta name="msapplication-TileColor" content="#603cba">
  <meta name="msapplication-TileImage" content="/favicons/mstile-144x144.png">
  <meta name="msapplication-config" content="/favicons/browserconfig.xml">
  <meta name="theme-color" content="#303284">

  <title>Apache License, Version 2.0</title>
  <link href='https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700%7cDroid+Serif:400,700'
rel='stylesheet' type='text/css'>
  <link href="/css/min.bootstrap.css" rel="stylesheet">
  <link href="/css/styles.css" rel="stylesheet">
```

<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at .

<http://www.apache.org/licenses/LICENSE-2.0> . Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License. -->

</head>

<body>

<!-- Navigation -->

<header>

<nav class="navbar navbar-default navbar-fixed-top">

<div class="container">

<div class="navbar-header">

<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">

Toggle navigation

</button>

</div>

<div class="collapse navbar-collapse" id="mainnav-collapse">

<div style="line-height:20px; padding-top:5px; float:left">Home » Licenses</div>

<ul class="nav navbar-nav navbar-right">

<li class="dropdown">

About

<ul class="dropdown-menu" role="menu">

Overview

Members

Process

Sponsorship

Glossary

FAQ

Contact

Projects

<li class="dropdown">

People

<ul class="dropdown-menu" role="menu">

Overview

Committers

Meritocracy

Roles

Planet Apache


```

<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Get Involved <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
    <li><a href="/foundation/getinvolved.html">Overview</a></li>
    <li><a href="http://community.apache.org/">Community Development</a></li>
    <li><a href="http://helpwanted.apache.org/">Help Wanted</a></li>
    <li><a href="http://www.apachecon.com/">ApacheCon</a></li>
  </ul>
</li>
<li><a href="/dyn/closer.cgi">Download</a></li>
<li class="dropdown">
  <a href="#" class="dropdown-toggle" data-toggle="dropdown">Support Apache <span
class="caret"></span></a>
  <ul class="dropdown-menu" role="menu">
    <li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
    <li><a href="/foundation/contributing.html">Donations</a></li>
    <li><a href="/foundation/buy_stuff.html">Buy Stuff</a></li>
    <li><a href="/foundation/thanks.html">Thanks</a></li>
  </ul>
</li>
</ul>
</div>
</div>
</nav>
</header>
<!-- / Navigation -->
<div class="container">
<div class="row">
<div class="col-md-9 col-sm-8 col-xs-12">
  
</div>
<div class="col-md-3 col-sm-4 col-xs-12">
  <div class="input-group" style="margin-bottom: 5px;">
    <script>
(function() {
var cx = '005703438322411770421:5mgshgrgx2u';
var gcse = document.createElement('script');
gcse.type = 'text/javascript';
gcse.async = true;
gcse.src = (document.location.protocol == 'https:' ? 'https:' : 'http:') +
  '//cse.google.com/cse.js?cx=' + cx;
var s = document.getElementsByTagName('script')[0];
s.parentNode.insertBefore(gcse, s);
})();
</script>
<gcse:searchbox-only></gcse:searchbox-only>
</div>

```



```

    <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/governance/">The Apache Way</a>
    <a role="button" class="btn btn-block btn-default btn-xs"
href="https://community.apache.org/contributors/">Contribute</a>
    <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/thanks.html">ASF Sponsors</a>
  </div>
</div>
</div>
<div class="container"><style type="text/css">
/* The following code is added by mdx_elementid.py
It was originally lifted from http://subversion.apache.org/style/site.css */
/*
* Hide class="elementid-permalink", except when an enclosing heading
* has the :hover property.
*/
.headerlink, .elementid-permalink {
visibility: hidden;
}
h2:hover > .headerlink, h3:hover > .headerlink, h1:hover > .headerlink, h6:hover > .headerlink, h4:hover >
.headerlink, h5:hover > .headerlink, dt:hover > .elementid-permalink { visibility: visible }</style>
<p>Apache License<br><br>Version 2.0, January 2004<br><br>
<a href="http://www.apache.org/licenses/">http://www.apache.org/licenses/</a> </p>
<p>TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION</p>
<p><strong><a name="definitions">1. Definitions</a></strong>.</p>
<p>"License" shall mean the terms and conditions for use, reproduction, and
distribution as defined by Sections 1 through 9 of this document.</p>
<p>"Licensor" shall mean the copyright owner or entity authorized by the
copyright owner that is granting the License.</p>
<p>"Legal Entity" shall mean the union of the acting entity and all other
entities that control, are controlled by, or are under common control with
that entity. For the purposes of this definition, "control" means (i) the
power, direct or indirect, to cause the direction or management of such
entity, whether by contract or otherwise, or (ii) ownership of fifty
percent (50%) or more of the outstanding shares, or (iii) beneficial
ownership of such entity.</p>
<p>"You" (or "Your") shall mean an individual or Legal Entity exercising
permissions granted by this License.</p>
<p>"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation source,
and configuration files.</p>
<p>"Object" form shall mean any form resulting from mechanical transformation
or translation of a Source form, including but not limited to compiled
object code, generated documentation, and conversions to other media types.</p>
<p>"Work" shall mean the work of authorship, whether in Source or Object form,
made available under the License, as indicated by a copyright notice that
is included in or attached to the work (an example is provided in the
Appendix below).</p>
<p>"Derivative Works" shall mean any work, whether in Source or Object form,
that is based on (or derived from) the Work and for which the editorial

```

revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

-

- You must give any other recipients of the Work or Derivative Works a copy of this License; and

You must cause any modified files to carry prominent notices stating that You changed the files; and

You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

<p>5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.</p>

<p>6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.</p>

<p>7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and

each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

```
Copyright [yyyy] [name of copyright owner]
```

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and

limitations under the License.

```
</pre></div></div>
```

```
<!-- Footer -->
```

```
<footer class="bg-primary">
```

```
<div class="container">
```

```
<div class="row">
```

```
<br />
```

```
<div class="col-sm-1">
```

```
</div>
```

```
<div class="col-sm-2">
```

```
<h5 class="white">Community</h5>
```

```
<ul class="list-unstyled white" role="menu">
```

```
<li><a href="http://community.apache.org/">Overview</a></li>
```

```
<li><a href="/foundation/conferences.html">Conferences</a></li>
```

```
<li><a href="http://community.apache.org/gsoc.html">Summer of Code</a></li>
```

```
<li><a href="http://community.apache.org/newcomers/">Getting Started</a></li>
```

```
<li><a href="/foundation/how-it-works.html">The Apache Way</a></li>
```

```
<li><a href="/travel/">Travel Assistance</a></li>
```

```
<li><a href="/foundation/getinvolved.html">Get Involved</a></li>
```

```
<li><a href="http://community.apache.org/newbiefaq.html">Community FAQ</a></li>
```

```
</ul>
```

```
</div>
```

```
<div class="col-sm-2">
```

```
<h5 class="white">Innovation</h5>
```

```
<ul class="list-unstyled white" role="menu">
```

```
<li><a href="http://incubator.apache.org/">Incubator</a></li>
```

```
<li><a href="http://labs.apache.org/">Labs</a></li>
```

```
<li><a href="/licenses/">Licensing</a></li>
```

```
<li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
```

```
<li><a href="/foundation/marks/">Trademark Policy</a></li>
```

```
<li><a href="/foundation/contact.html">Contacts</a></li>
```

```
</ul>
```

```
</div>
```

```
<div class="col-sm-2">
```

```
<h5 class="white">Tech Operations</h5>
```

```
<ul class="list-unstyled white" role="menu">
```

```
<li><a href="/dev/">Developer Information</a></li>
```

```
<li><a href="/dev/infrastructure.html">Infrastructure</a></li>
```

```
<li><a href="/security/">Security</a></li>
```

```
<li><a href="http://status.apache.org">Status</a></li>
```

```
<li><a href="/foundation/contact.html">Contacts</a></li>
```


</div>

<div class="col-sm-2">

<h5 class="white">Press</h5>

<ul class="list-unstyled white" role="menu">

Overview

ASF News

Announcements

Twitter Feed

Contacts

</div>

<div class="col-sm-2">

<h5 class="white">Legal</h5>

<ul class="list-unstyled white" role="menu">

Legal Affairs

Licenses

Trademark Policy

Public Records

Privacy Policy

Export Information

License/Distribution FAQ

Contacts

</div>

<div class="col-sm-1">

</div>

</div>

<hr class="col-lg-12 hr-white" />

<div class="row">

<div class="col-lg-12">

<p class="text-center">Copyright © 2016 The Apache Software Foundation, Licensed under the Apache License, Version 2.0.</p>

<p class="text-center">Apache and the Apache feather logo are trademarks of The Apache Software Foundation.</p>

</div>

</div>

</div>

</footer>

<!-- / Footer -->

<script src="/js/jquery-2.1.1.min.js"></script>

<script src="/js/bootstrap.js"></script>

</body>

</html>

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of,

the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This copy of Jackson JSON processor annotations is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

This copy of Jackson JSON processor streaming parser/generator is licensed under the
Apache (Software) License, version 2.0 ("the License").
See the License for details about distribution rights, and the
specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>301 Moved Permanently</title>
</head><body>
<h1>Moved Permanently</h1>
<p>The document has moved <a href="https://opensource.org/licenses/mit-license.php">here</a>.</p>
</body></html>
```

Format: <http://www.debian.org/doc/packaging-manuals/copyright-format/1.0/>

Upstream-Name: schema-registry

Source: <https://github.com/confluentinc/schema-registry>

Files: *

Copyright: 2018 Confluent, Inc.

License: Apache-2

Files: core/*

Copyright: 2015 Confluent, Inc.

License: Confluent Community License

License: Confluent Community License

Licensed under the Confluent Community License; you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.confluent.io/confluent-community-license>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

License: Apache-2

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

On Debian systems, the Apache 2.0 license can be found in `/usr/share/common-licenses/Apache-2.0`.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the

Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside

or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer,

and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This distribution has a binary dependency on jersey, which is available under the CDDL License as described below.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL - Version 1.1)

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

- 1.4. Executable means the Covered Software in any form other than Source Code.
- 1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
- 1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
- 1.7. License means this document.
- 1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
- 1.9. Modifications means the Source Code and Executable form of any of the following:
- A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
 - B. Any new file that contains any part of the Original Software or previous Modification; or
 - C. Any new file that is contributed or otherwise made available under the terms of this License.
- 1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
- 1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
- 1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
- 1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use,

practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to

grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of

the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING

NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The following commands were used to generate license and notice files. Replace <VERSION> with the Schema Registry version, <SRC_PATH> with the path to the Schema Registry source directory, and <LICENSE_TOOL_PATH> with the path of the license tool.

```
cd <SRC_PATH>
mvn package -DskipTests
mkdir /tmp/jars
mkdir /tmp/overrides
cp package-schema-registry/target/kafka-schema-registry-package-<VERSION>-package/share/java/schema-registry/*.jar /tmp/jars/
cp package-kafka-serde-tools/target/kafka-serde-tools-package-<VERSION>-package/share/java/kafka-serde-tools/*.jar /tmp/jars/
cd <LICENSE_TOOL_PATH>
./bin/run_license_job.bash -i /tmp/jars -l <SRC_PATH>/licenses -n <SRC_PATH>/notices -h <SRC_PATH>/licenses-and-notices.html -o /tmp/overrides
```

Apache Avro
Copyright 2009-2014 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

```
<html lang="en-US">
<head>
<title>GlassFish Dual License Header and License Notice GPL v2 and CDDL</title>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<style type="text/css">
li { display: block;}
</style>
</head>

<body leftmargin="0" topmargin="0" marginheight="0" marginwidth="0" bgcolor="#ffffff">

<h3>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0</h3>

<ul>
<li><p><b>1. Definitions.</b></p>
<ul>
<li>
<p>
<b>1.1. &ldquo;Contributor&rdquo;</b> means each individual or
entity that creates or contributes to the creation of
Modifications.
</p>
</li>
</ul>
</li>
```


<p>
1.2. “Contributor Version” means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
</p>

<p>
1.3. “Covered Software” means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
</p>

<p>
1.4. “Executable” means the Covered Software in any form other than Source Code.
</p>

<p>
1.5. “Initial Developer” means the individual or entity that first makes Original Software available under this License.
</p>

<p>
1.6. “Larger Work” means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
</p>

<p>
1.7. “License” means this document.
</p>

<p>
1.8. “Licensable” means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
</p>

<p>
1.9. “Modifications” means the Source Code and Executable form of any of the following:

</p>

<p>
A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

</p>

<p>
B. Any new file that contains any part of the Original Software or previous Modification; or

</p>

<p>
C. Any new file that is contributed or otherwise made available under the terms of this License.

</p>

<p>
1.10. “Original Software” means the Source Code and Executable form of computer software code that is originally released under this License.
</p>

<p>
1.11. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

</p>

<p>
1.12. “Source Code” means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

</p>

<p>
1.13. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

</p>

<p>2. License Grants.</p>

<p>2.1. The Initial Developer Grant.</p>

<p>

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

</p>

<p>
(a) under intellectual property rights
(other than patent or trademark) Licensable by Initial
Developer, to use, reproduce, modify, display, perform,
sublicense and distribute the Original Software (or
portions thereof), with or without Modifications, and/or
as part of a Larger Work; and
</p>

<p>
(b) under Patent Claims infringed by the
making, using or selling of Original Software, to make,
have made, use, practice, sell, and offer for sale,
and/or otherwise dispose of the Original Software (or
portions thereof).
</p>

<p>
(c) The licenses granted in Sections 2.1(a)
and (b) are effective on the date Initial Developer first
distributes or otherwise makes the Original Software
available to a third party under the terms of this License.
</p>

<p>
(d) Notwithstanding Section 2.1(b)
above, no patent license is granted: (1) for code
that You delete from the Original Software, or
(2) for infringements caused by: (i) the
modification of the Original Software, or (ii) the
combination of the Original Software with other software
or devices.
</p>

<p>2.2. Contributor Grant.</p>
<p>
Conditioned upon Your compliance with Section 3.1 below
and subject to third party intellectual property claims,
each Contributor hereby grants You a world-wide,
royalty-free, non-exclusive license:
</p>

<p>
(a) under intellectual property rights
(other than patent or trademark) Licensable by
Contributor to use, reproduce, modify, display, perform,
sublicense and distribute the Modifications created by
such Contributor (or portions thereof), either on an
unmodified basis, with other Modifications, as Covered
Software and/or as part of a Larger Work; and
</p>

<p>
(b) under Patent Claims infringed by the
making, using, or selling of Modifications made by that
Contributor either alone and/or in combination with its
Contributor Version (or portions of such combination),
to make, use, sell, offer for sale, have made, and/or
otherwise dispose of: (1) Modifications made by
that Contributor (or portions thereof); and (2) the
combination of Modifications made by that Contributor
with its Contributor Version (or portions of such
combination).
</p>

<p>
(c) The licenses granted in Sections 2.2(a)
and 2.2(b) are effective on the date Contributor first
distributes or otherwise makes the Modifications available
to a third party.
</p>

<p>
(d) Notwithstanding Section 2.2(b)
above, no patent license is granted: (1) for any
code that Contributor has deleted from the Contributor
Version; (2) for infringements caused by:
(i) third party modifications of Contributor
Version, or (ii) the combination of Modifications
made by that Contributor with other software (except as
part of the Contributor Version) or other devices; or
(3) under Patent Claims infringed by Covered
Software in the absence of Modifications made by that
Contributor.
</p>

<p>3. Distribution Obligations.</p>

<p>3.1. Availability of Source Code.</p>

<p>

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

</p>

<p>3.2. Modifications.</p>

<p>

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

</p>

<p>3.3. Required Notices.</p>

<p>

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

</p>

<p>3.4. Application of Additional Terms.</p>

<p>

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

</p>

<p>3.5. Distribution of Executable Versions.</p>

<p>

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

</p>

<p>3.6. Larger Works.</p>

<p>

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

</p>

<p>4. Versions of the License.</p>

<p>4.1. New Versions.</p>

<p>

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

</p>

<p>4.2. Effect of New Versions.</p>

<p>

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

</p>

<p>4.3. Modified Versions.</p>

<p>

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

</p>

<p>5. DISCLAIMER OF WARRANTY.</p>

<p>
COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN
“AS IS” BASIS, WITHOUT WARRANTY OF ANY KIND,
EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION,
WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS,
MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT.
THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE
COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE
PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER
OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY
SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY
CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY
COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS
DISCLAIMER.

</p>

<p>6. TERMINATION.</p>

<p>
6.1. This License and the rights granted
hereunder will terminate automatically if You fail to comply
with terms herein and fail to cure such breach within 30
days of becoming aware of the breach. Provisions which, by
their nature, must remain in effect beyond the termination
of this License shall survive.

</p>

<p>
6.2. If You assert a patent infringement claim
(excluding declaratory judgment actions) against Initial Developer
or a Contributor (the Initial Developer or Contributor against whom
You assert such claim is referred to as “Participant”)
alleging that the Participant Software (meaning the Contributor
Version where the Participant is a Contributor or the Original Software
where the Participant is the Initial Developer) directly or indirectly
infringes any patent, then any and all rights granted directly or
indirectly to You by such Participant, the Initial Developer (if the

Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

</p>

<p>

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

</p>

7. LIMITATION OF LIABILITY.

<p>

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

</p>

8. U.S. GOVERNMENT END USERS.

<p>

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995),

consisting of "commercial computer software" (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

</p>

<p>9. MISCELLANEOUS.</p>

<p>

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

</p>

<p>10. RESPONSIBILITY FOR CLAIMS.</p>

<p>

As between Initial Developer and the Contributors, each

party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

</p>

<p>

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

</p>

<p>

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions).

Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

</p>

<h3>The GNU General Public License (GPL) Version 2, June 1991</h3>

<p>

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

</p>

<p>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

</p>

<p>Preamble</p>

<p>

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it

to your programs, too.

</p>

<p>

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

</p>

<p>

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

</p>

<p>

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

</p>

<p>

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

</p>

<p>

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

</p>

<p>

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

</p>

<p>

The precise terms and conditions for copying, distribution and modification follow.

</p>

<p>TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION</p>

<ul style="margin-left:0; padding-left:0; border-left:0">

<p>

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

</p>

<p>

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

</p>

<p>

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

</p>

<p>

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

</p>

<p>

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

</p>

<p>

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

</p>

<p>

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

</p>

<p>

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

</p>

<p>

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

</p>

<p>

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

</p>

<p>

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of

a storage or distribution medium does not bring the other work under the scope of this License.

</p>

<p>

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

</p>

<p>

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

</p>

<p>

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

</p>

<p>

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

</p>

<p>

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the

major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

</p>

<p>

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

</p>

<p>

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

</p>

<p>

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

</p>

<p>

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

</p>

<p>

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

</p>

<p>

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

</p>

<p>

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

</p>

<p>

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

</p>

<p>

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

</p>

<p>

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

</p>

<p>

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

</p>

<p>

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

</p>

<p>NO WARRANTY</p>

<p>

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

</p>

<p>
12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

</p>

<p>END OF TERMS AND CONDITIONS</p>

<p>How to Apply These Terms to Your New Programs</p>

<p>

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

</p>

<p>

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

</p>

<p>

One line to give the program's name and a brief idea of what it does.

</p>

<p>

Copyright (C) <year> <name of author>

</p>

<p>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

</p>

<p>

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

</p>

<p>

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

</p>

<p>

Also add information on how to contact you by electronic and paper mail.

</p>

<p>

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

</p>

<p>

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

</p>

<p>

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

</p>

<p>

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

</p>

<p>

Yoyodyne, Inc., hereby disclaims all copyright interest in the program
'Gnomovision' (which makes passes at compilers) written by James Hacker.

</p>

<p>

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

</p>

<p>

This General Public License does not permit incorporating your program into
proprietary programs. If your program is a subroutine library, you may
consider it more useful to permit linking proprietary applications with
the library. If this is what you want to do, use the GNU Library General
Public License instead of this License.

</p>

<li style="background-color:yellow;">

<p>"CLASSPATH" EXCEPTION TO THE GPL VERSION 2</p>

<p>

Certain source files distributed by Sun Microsystems, Inc. are subject to
the following clarification and special exception to the GPL Version 2, but
only where Sun has expressly included in the particular source file's
header the words

"Sun designates this particular file as subject to the "Classpath"
exception as provided by Sun in the License file that accompanied this code."

</p>

<p>

Linking this library statically or dynamically with other modules is making
a combined work based on this library. Thus, the terms and conditions of the
GNU General Public License Version 2 cover the whole combination.

</p>

<p>

As a special exception, the copyright holders of this library give you
permission to link this library with independent modules to produce an
executable, regardless of the license terms of these independent modules,
and to copy and distribute the resulting executable under terms of your
choice, provided that you also meet, for each linked independent module,
the terms and conditions of the license of that module.? An independent
module is a module which is not derived from or based on this library.? If
you modify this library, you may extend this exception to your version of
the library, but you are not obligated to do so.? If you do not wish to do
so, delete this exception statement from your version.

</p>


```

</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Untitled Document</title>
</head>

<body>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<!-- @page { margin: 0.79in } P { margin-bottom: 0.08in } H1 { margin-top: 0in; margin-bottom: 0.17in; widows:
0; orphans: 0 } H1.western { font-family: "Times New Roman", serif; font-size: 12pt; font-weight: normal } H1.cjk
{ font-family: "Andale Sans UI"; font-size: 12pt; font-weight: normal } H1.clt { font-family: "Tahoma"; font-size:
12pt; font-weight: normal } -->
<BODY DIR="LTR">
<p><strong>COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)Version 1.1</strong></p>
<p>1. Definitions.</p>
<blockquote>1.1. &ldquo;Contributor&rdquo; means each individual or entity that creates or contributes to the
creation of Modifications.</blockquote>
<blockquote>1.2. &ldquo;Contributor Version&rdquo; means the combination of the Original Software, prior
Modifications used by a Contributor (if any), and the Modifications made by that particular
Contributor.</blockquote>
<blockquote>1.3. &ldquo;Covered Software&rdquo; means (a) the Original Software, or (b) Modifications, or (c)
the combination of files containing Original Software with files containing Modifications, in each case including
portions thereof.</blockquote>
<blockquote>1.4. &ldquo;Executable&rdquo; means the Covered Software in any form other than Source
Code.</blockquote>
<blockquote>1.5. &ldquo;Initial Developer&rdquo; means the individual or entity that first makes Original
Software available under this License.</blockquote>
<blockquote>1.6. &ldquo;Larger Work&rdquo; means a work which combines Covered Software or portions
thereof with code not governed by the terms of this License.</blockquote>
<blockquote>1.7. &ldquo;License&rdquo; means this document.</blockquote>
<blockquote>1.8. &ldquo;Licensable&rdquo; means having the right to grant, to the maximum extent possible,
whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed
herein.</blockquote>
<blockquote>1.9. &ldquo;Modifications&rdquo; means the Source Code and Executable form of any of the
following:</blockquote>
<blockquote>A. Any file that results from an addition to, deletion from or modification of the contents of a file
containing Original Software or previous Modifications;</blockquote>
<blockquote>B. Any new file that contains any part of the Original Software or previous Modification;
or</blockquote>
<blockquote>C. Any new file that is contributed or otherwise made available under the terms of this
License.</blockquote>
<blockquote>1.10. &ldquo;Original Software&rdquo; means the Source Code and Executable form of computer

```

software code that is originally released under this License.</blockquote>

<blockquote>1.11. “Patent Claims” means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.</blockquote>

<blockquote>1.12. “Source Code” means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.</blockquote>

<blockquote>1.13. “You” (or “Your”) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, “You” includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, “control” means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.</blockquote>

<p>2. License Grants.</p>

<blockquote>2.1. The Initial Developer Grant.</blockquote>

<blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:</blockquote>

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).</blockquote>

<blockquote>(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.</blockquote>

<blockquote>(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.</blockquote>

<blockquote>2.2. Contributor Grant.</blockquote>

<blockquote>Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:</blockquote>

<blockquote>(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and</blockquote>

<blockquote>(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).</blockquote>

<blockquote>(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.</blockquote>

<blockquote>(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered

Software in the absence of Modifications made by that Contributor.</blockquote>

<p>3. Distribution Obligations.</p>

<blockquote>3.1. Availability of Source Code.</blockquote>

<blockquote>Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.</blockquote>

<blockquote>3.2. Modifications.</blockquote>

<blockquote>The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.</blockquote>

<blockquote>3.3. Required Notices.</blockquote>

<blockquote>You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.</blockquote>

<blockquote>3.4. Application of Additional Terms.</blockquote>

<blockquote>You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.</blockquote>

<blockquote>3.5. Distribution of Executable Versions.</blockquote>

<blockquote>You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.</blockquote>

<blockquote>3.6. Larger Works.</blockquote>

<blockquote>You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.</blockquote>

<p>4. Versions of the License.</p>

<blockquote>4.1. New Versions.</blockquote>

<blockquote>Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.</blockquote>

<blockquote>4.2. Effect of New Versions.</blockquote>

<blockquote>You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial

Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.</blockquote>

<blockquote>4.3. Modified Versions.</blockquote>

<blockquote>When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.</blockquote>

<p>5. DISCLAIMER OF WARRANTY.</p>

<blockquote>

<p>COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.</p>

</blockquote>

<p>6. TERMINATION.</p>

<blockquote>6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.</blockquote>

<blockquote>6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.</blockquote>

<blockquote>6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.</blockquote>

<blockquote>6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.</blockquote>

<p>7. LIMITATION OF LIABILITY.</p>

<blockquote>

<p>UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING

NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. § 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.

59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies

of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program" below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as

"you".</p>

<p>Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program).

Whether that is true depends on what the Program does.</p>

<p>1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.</p>

<p>You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.</p>

<p>2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:</p>

<blockquote>a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.</blockquote>

<blockquote>b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.</blockquote>

<blockquote>c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)</blockquote>

<p>These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.</p>

<p>Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.</p>

<p>In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.</p>

<p>3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:</p>

<blockquote>a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,</blockquote>

<blockquote>b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,</blockquote>

<blockquote>c) Accompany it with the information you received as to the offer to distribute corresponding source

code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

```
<blockquote>One line to give the program's name and a brief idea of what it does.<br />
Copyright (C) &lt;year&gt; &lt;name of author&gt;</blockquote>
```

```
<blockquote>This program is free software; you can redistribute it and/or modify it under the terms of the GNU
General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your
option) any later version.</blockquote>
```

```
<blockquote>This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY;
without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See
the GNU General Public License for more details.</blockquote>
```

```
<blockquote>You should have received a copy of the GNU General Public License along with this program; if not,
```

write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307

USA</blockquote>

<p>Also add information on how to contact you by electronic and paper mail.</p>

<p>If the program is interactive, make it output a short notice like this when it starts in an interactive mode:</p>

<blockquote>Gnomovision version 69, Copyright (C) year name of author Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.</blockquote>

<p>The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.</p>

<p>You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:</p>

<blockquote>Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.</blockquote>

<blockquote>signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice</blockquote>

<p>This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.</p>

<li style="background-color:yellow;">

<p>"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.</p>

<p>

</p>

<p>

</p>

</body>

</html>

This copy of Jackson JSON processor is licensed under the

Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/>

A copy is also included with both the the downloadable source code package and jar that contains class bytecodes, as file "ASL 2.0". In both cases, that file should be located next to this file: in source distribution the location should be "release-notes/asl"; and in jar "META-INF/"

The project is licensed under the Confluent Community License, except for client libs, which is under the Apache 2.0 license.

See LICENSE file in each subfolder for detailed license agreement.

Jackson JSON processor

Jackson is a high-performance, Free/Open Source JSON processing library.

It was originally written by Tatu Saloranta (tatu.saloranta@iki.fi), and has been in development since 2007.

It is currently developed by a community of developers, as well as supported commercially by FasterXML.com.

Licensing

Jackson core and extension components may be licensed under different licenses.

To find the details that apply to this artifact see the accompanying LICENSE file.

For more information, including possible other licensing options, contact FasterXML.com (<http://fasterxml.com>).

Credits

A list of contributors may be found from CREDITS file, which is included in some artifacts (usually source distributions); but is always available from the source code management (SCM) system project uses.

Apache log4j

Copyright 2007 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

```
<!DOCTYPE html>
```

```
<html lang="en">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<meta name="description" content="Home page of The Apache Software Foundation">
```

```
<link rel="apple-touch-icon" sizes="57x57" href="/favicons/apple-touch-icon-57x57.png">
<link rel="apple-touch-icon" sizes="60x60" href="/favicons/apple-touch-icon-60x60.png">
<link rel="apple-touch-icon" sizes="72x72" href="/favicons/apple-touch-icon-72x72.png">
<link rel="apple-touch-icon" sizes="76x76" href="/favicons/apple-touch-icon-76x76.png">
<link rel="apple-touch-icon" sizes="114x114" href="/favicons/apple-touch-icon-114x114.png">
<link rel="apple-touch-icon" sizes="120x120" href="/favicons/apple-touch-icon-120x120.png">
<link rel="apple-touch-icon" sizes="144x144" href="/favicons/apple-touch-icon-144x144.png">
<link rel="apple-touch-icon" sizes="152x152" href="/favicons/apple-touch-icon-152x152.png">
<link rel="apple-touch-icon" sizes="180x180" href="/favicons/apple-touch-icon-180x180.png">
<link rel="icon" type="image/png" href="/favicons/favicon-32x32.png" sizes="32x32">
<link rel="icon" type="image/png" href="/favicons/favicon-194x194.png" sizes="194x194">
<link rel="icon" type="image/png" href="/favicons/favicon-96x96.png" sizes="96x96">
<link rel="icon" type="image/png" href="/favicons/android-chrome-192x192.png" sizes="192x192">
<link rel="icon" type="image/png" href="/favicons/favicon-16x16.png" sizes="16x16">
<link rel="manifest" href="/favicons/manifest.json">
<link rel="shortcut icon" href="/favicons/favicon.ico">
<meta name="msapplication-TileColor" content="#603cba">
<meta name="msapplication-TileImage" content="/favicons/mstile-144x144.png">
<meta name="msapplication-config" content="/favicons/browserconfig.xml">
<meta name="theme-color" content="#303284">
```

```
<title>Licenses</title>
```

```
<link href='https://fonts.googleapis.com/css?family=Source+Sans+Pro:400,700%7cDroid+Serif:400,700'
rel='stylesheet' type='text/css'>
```

```
<link href="/css/min.bootstrap.css" rel="stylesheet">
```

```
<link href="/css/styles.css" rel="stylesheet">
```

```
<!-- Licensed to the Apache Software Foundation (ASF) under one or more contributor license agreements. See
the NOTICE file distributed with this work for additional information regarding copyright ownership. The ASF
licenses this file to you under the Apache License, Version 2.0 (the "License"); you may not use this file
except in compliance with the License. You may obtain a copy of the License at
http://www.apache.org/licenses/LICENSE-2.0 . Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR
CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing
permissions and limitations under the License. -->
```

```
</head>
```

```
<body>
```

```
<!-- Navigation -->
```

```
<header>
```

```
<nav class="navbar navbar-default navbar-fixed-top">
```

```
<div class="container">
```

```
<div class="navbar-header">
```

```
<button class="navbar-toggle" type="button" data-toggle="collapse" data-target="#mainnav-collapse">
```

```
<span class="sr-only">Toggle navigation</span>
```

```
<span class="icon-bar"></span>
```



```

    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
  </button>
  <a href="#" class="navbar-brand"><span class="glyphicon glyphicon-home"></span></a>
</div>
<div class="collapse navbar-collapse" id="mainnav-collapse">
  <div style="line-height:20px; padding-top:5px; float:left"><a href="/">Home</a>&nbsp;&raquo;&nbsp;&nbsp;<a
href="/licenses/">Licenses</a></div>
  <ul class="nav navbar-nav navbar-right">
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown">About <span class="caret"></span></a>
      <ul class="dropdown-menu" role="menu">
        <li><a href="/foundation">Overview</a></li>
        <li><a href="/foundation/members.html">Members</a></li>
        <li><a href="/foundation/how-it-works.html">Process</a></li>
        <li><a href="/foundation/sponsorship.html">Sponsorship</a></li>
        <li><a href="/foundation/glossary.html">Glossary</a></li>
        <li><a href="/foundation/preFAQ.html">FAQ</a></li>
        <li><a href="/foundation/contact.html ">Contact</a></li>
      </ul>
    </li>
    <li><a href="/index.html#projects-list">Projects</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown">People <span class="caret"></span></a>
      <ul class="dropdown-menu" role="menu">
        <li><a href="http://people.apache.org/">Overview</a></li>
        <li><a href="http://people.apache.org/committer-index.html">Committers</a></li>
        <li><a href="/foundation/how-it-works.html#meritocracy">Meritocracy</a></li>
        <li><a href="/foundation/how-it-works.html#roles">Roles</a></li>
        <li><a href="http://planet.apache.org/">Planet Apache</a></li>
      </ul>
    </li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown">Get Involved <span
class="caret"></span></a>
      <ul class="dropdown-menu" role="menu">
        <li><a href="/foundation/getinvolved.html">Overview</a></li>
        <li><a href="http://community.apache.org/">Community Development</a></li>
        <li><a href="http://helpwanted.apache.org/">Help Wanted</a></li>
        <li><a href="http://www.apachecon.com/">ApacheCon</a></li>
      </ul>
    </li>
    <li><a href="/dyn/closer.cgi">Download</a></li>
    <li class="dropdown">
      <a href="#" class="dropdown-toggle" data-toggle="dropdown">Support Apache <span
class="caret"></span></a>
      <ul class="dropdown-menu" role="menu">
        <li><a href="/foundation/sponsorship.html">Sponsorship</a></li>

```

```

        <li><a href="/foundation/contributing.html">Donations</a></li>
        <li><a href="/foundation/buy_stuff.html">Buy Stuff</a></li>
        <li><a href="/foundation/thanks.html">Thanks</a></li>
    </ul>
</li>
</ul>
</div>
</div>
</nav>
</header>
<!-- / Navigation -->
<div class="container">
    <div class="row">
        <div class="col-md-9 col-sm-8 col-xs-12">
            
        </div>
        <div class="col-md-3 col-sm-4 col-xs-12">
            <div class="input-group" style="margin-bottom: 5px;">
                <script>
(function() {
    var cx = '005703438322411770421:5mgshgrgx2u';
    var gcse = document.createElement('script');
    gcse.type = 'text/javascript';
    gcse.async = true;
    gcse.src = (document.location.protocol == 'https:' ? 'https:' : 'http:') +
        '//cse.google.com/cse.js?cx=' + cx;
    var s = document.getElementsByTagName('script')[0];
    s.parentNode.insertBefore(gcse, s);
})();
</script>
        <gcse:searchbox-only></gcse:searchbox-only>
    </div>
    <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/governance/">The Apache Way</a>
    <a role="button" class="btn btn-block btn-default btn-xs"
href="https://community.apache.org/contributors/">Contribute</a>
    <a role="button" class="btn btn-block btn-default btn-xs" href="/foundation/thanks.html">ASF Sponsors</a>
    </div>
</div>
</div>
<div class="container"><style type="text/css">
/* The following code is added by mdx_elementid.py
   It was originally lifted from http://subversion.apache.org/style/site.css */
/*
* Hide class="elementid-permalink", except when an enclosing heading
* has the :hover property.
*/
.headerlink, .elementid-permalink {
    visibility: hidden;

```

```

}
h2:hover > .headerlink, h3:hover > .headerlink, h1:hover > .headerlink, h6:hover > .headerlink, h4:hover >
.headerlink, h5:hover > .headerlink, dt:hover > .elementid-permalink { visibility: visible }</style>
<p>The Apache Software Foundation uses various licenses to <a href="#distributions">distribute
software and documentation</a>, to accept regular
<a href="#clas">contributions from individuals and corporations</a>, and to accept
larger <a href="#grants">grants of existing software products</a>.</p>
<p>These licenses help us achieve our goal of providing reliable and
long-lived software products through collaborative open source software
development. In all cases, contributors retain full rights to use their
original contributions for any other purpose outside of Apache while
providing the ASF and its projects the right to distribute and build upon
their work within Apache.</p>
<h1 id="distributions">Licensing of Distributions<a class="headerlink" href="#distributions" title="Permanent
link">&para;</a></h1>
<p>All software produced by The Apache Software Foundation or any of its
projects or subjects is licensed according to the terms of the documents
listed below.</p>
<h3 id="2.0">Apache License, Version 2.0 (current)<a class="headerlink" href="#2.0" title="Permanent
link">&para;</a></h3>
<p><a href="LICENSE-2.0">http://www.apache.org/licenses/LICENSE-2.0</a> (
<a href="LICENSE-2.0.txt">TXT</a> or <a href="LICENSE-2.0.html">HTML</a> )</p>
<p>The 2.0 version of the Apache License was approved by the ASF in 2004. The
goals of this license revision have been to reduce the number of frequently
asked questions, to allow the license to be reusable without modification
by any project (including non-ASF projects), to allow the license to be
included by reference instead of listed in every file, to clarify the
license on submission of contributions, to require a patent license on
contributions that necessarily infringe the contributor's own patents, and
to move comments regarding Apache and other inherited attribution notices
to a location outside the license terms (the <a href="example-NOTICE.txt">NOTICE
file</a> ).</p>
<p>The result is a license that is supposed to be compatible with other open
source licenses, while remaining true to the original goals of the Apache
Group and supportive of collaborative development across both nonprofit and
commercial organizations. The Apache Software Foundation is still trying to
determine if this version of the Apache License is <a href="GPL-compatibility.html">compatible with the
GPL</a>.</p>
<p>All packages produced by the ASF are implicitly licensed under the Apache
License, Version 2.0, unless otherwise explicitly stated. More developer
documentation on how to apply the Apache License to your work can be found
in * <a href="..dev/apply-license.html">Applying the Apache License, Version 2.0</a>
*.</p>
<h3 id="1.1">Apache License, Version 1.1 (historic)<a class="headerlink" href="#1.1" title="Permanent
link">&para;</a></h3>
<p><a href="LICENSE-1.1">http://www.apache.org/licenses/LICENSE-1.1</a> </p>
<p>The 1.1 version of the Apache License was approved by the ASF in 2000. The
primary change from the 1.0 license is in the 'advertising clause' (section

```

3 of the 1.0 license); derived products are no longer required to include attribution in their advertising materials, only in their documentation.</p>

<p>Individual packages licensed under the 1.1 version may have used different wording due to varying requirements for attribution or mark identification, but the binding terms were all the same.</p>

<h3 id="1.0">Apache License, Version 1.0 (historic)¶</h3>

<p>http://www.apache.org/licenses/LICENSE-1.0 </p>

<p>This is the original Apache License which applies only to older versions of Apache packages (such as version 1.2 of the Web server).</p>

<h1 id="clas">Contributor License Agreements¶</h1>

<p>The ASF desires that all contributors of ideas, code, or documentation to any Apache projects complete, sign, and submit (via fax or email) an Individual Contributor License Agreement (ICLA).

The purpose of this agreement is to clearly define the terms under which intellectual property has been contributed to the ASF and thereby allow us to defend the project should there be a legal dispute regarding the software at some future time. A signed ICLA is required to be on file before an individual is given commit rights to an ASF project.</p>

<p>For a corporation that has assigned employees to work on an Apache project, a Corporate CLA (CCLA) is available for contributing intellectual property via the corporation, that may have been assigned as part of an employment agreement. Note that a Corporate CLA does not remove the need for every developer to sign their own ICLA as an individual, to cover any of their contributions which are not owned by the corporation signing the CCLA.</p>

<p>The ICLA is not tied to any employer you may have, so it is recommended to use one's personal email address in the contact details, rather than an @work address.</p>

<p>Your Full name will be published unless you provide an alternative Public name. For example if your full name is Andrew Bernard Charles Dickens, but you wish to be known as Andrew Dickens, please enter the latter as your Public name.</p>

<p>The email address and other contact details are not published.</p>

<h1 id="grants">Software Grants¶</h1>

<p>When an individual or corporation decides to donate a body of existing software or documentation to one of the Apache projects, they need to execute a formal Software Grant Agreement (SGA) with the ASF. Typically, this is done after negotiating approval with the ASF Incubator or one of the PMCs, since the ASF will not accept software unless there is a viable community available to support a collaborative project.</p>

<h1 id="submitting">Submitting License Agreements and Grants¶</h1>

<p>Documents may be submitted by fax or email.</p>

<p>If submitting by fax, please print, sign, and fax all pages of the document to +1-919-573-9199. Please send documents right-side-up, first page first; and send only one document per fax.</p>

<p>If submitting by email, please fill the form with a pdf viewer, print, sign, scan all pages into a single pdf file, and email the pdf file as an attachment to secretary@apache.org. If possible, send the attachment from the email address in the document.

Please send only one document per email.</p>

<p>If you prefer to sign electronically, please fill the form, save it locally (e.g. icla.pdf), and sign the file by preparing a detached PGP signature. For example,</p>

<blockquote>

<p>gpg --armor --detach-sign icla.pdf</p>

</blockquote>

<p>The above will create a file icla.pdf.asc. Send both the file and signature as attachments in the same email to secretary@apache.org. Please send only one document (file plus signature) per email. Please do not submit your public key to Apache. Instead, please upload your public key to pgpkeys.mit.edu. </p>

<p>The files typically are named

icla.pdf and icla.pdf.asc for individual agreements;

ccla.pdf and ccla.pdf.asc for corporate agreements;

software-grant.pdf and software-grant.pdf.asc for grants. </p>

<h1 id="crypto">Export restrictions¶</h1>

<p>For export restriction information, please consult our ASF Export Classifications page.</p>

<h1 id="trademarks">Trademark and Logo Usage¶</h1>

<p>For ASF trademark and logo usage information, please consult our ASF Trademark Use Policy page.</p>

<h1 id="questions">Questions?¶</h1>

<p>For answers to frequently asked licensing questions, please consult our

Licensing Frequently Asked Questions page.</p></div>

<!-- Footer -->

<footer class="bg-primary">

<div class="container">

<div class="row">

<div class="col-sm-1">

</div>

<div class="col-sm-2">

<h5 class="white">Community</h5>

<ul class="list-unstyled white" role="menu">

Overview

Conferences

Summer of Code

Getting Started

The Apache Way

Travel Assistance

Get Involved

```
<li><a href="http://community.apache.org/newbiefaq.html">Community FAQ</a></li>
</ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Innovation</h5>
  <ul class="list-unstyled white" role="menu">
<li><a href="http://incubator.apache.org/">Incubator</a></li>
<li><a href="http://labs.apache.org/">Labs</a></li>
<li><a href="/licenses/">Licensing</a></li>
<li><a href="/foundation/license-faq.html">Licensing FAQ</a></li>
<li><a href="/foundation/marks/">Trademark Policy</a></li>
<li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Tech Operations</h5>
  <ul class="list-unstyled white" role="menu">
<li><a href="/dev/">Developer Information</a></li>
<li><a href="/dev/infrastructure.html">Infrastructure</a></li>
<li><a href="/security/">Security</a></li>
<li><a href="http://status.apache.org">Status</a></li>
<li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Press</h5>
  <ul class="list-unstyled white" role="menu">
<li><a href="/press/">Overview</a></li>
<li><a href="https://blogs.apache.org/">ASF News</a></li>
<li><a href="https://blogs.apache.org/foundation/">Announcements</a></li>
<li><a href="https://twitter.com/TheASF">Twitter Feed</a></li>
<li><a href="/press/#contact">Contacts</a></li>
  </ul>
</div>
```

```
<div class="col-sm-2">
<h5 class="white">Legal</h5>
  <ul class="list-unstyled white" role="menu">
<li><a href="/legal/">Legal Affairs</a></li>
<li><a href="/licenses/">Licenses</a></li>
<li><a href="/foundation/marks/">Trademark Policy</a></li>
<li><a href="/foundation/records/">Public Records</a></li>
  <li><a href="/foundation/policies/privacy.html">Privacy Policy</a></li>
<li><a href="/licenses/exports/">Export Information</a></li>
<li><a href="/foundation/license-faq.html">License/Distribution FAQ</a></li>
  </ul>
</div>
```

```

    <li><a href="/foundation/contact.html">Contacts</a></li>
  </ul>
</div>

<div class="col-sm-1">
</div>

</div>
<hr class="col-lg-12 hr-white" />
<div class="row">
  <div class="col-lg-12">
    <p class="text-center">Copyright © 2016 The Apache Software Foundation, Licensed under the <a
class="white" href="http://www.apache.org/licenses/LICENSE-2.0">Apache License, Version 2.0</a>.</p>
    <p class="text-center">Apache and the Apache feather logo are trademarks of The Apache Software
Foundation.</p>
  </div>
</div>
</div>

</footer>

<!-- / Footer -->

<script src="/js/jquery-2.1.1.min.js"></script>
<script src="/js/bootstrap.js"></script>
</body>
</html>
Apache License
    Version 2.0, January 2004
    http://www.apache.org/licenses/

```

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of,

publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution

notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

This product currently only contains code developed by authors of specific components, as identified by the source code files; if such notes are missing files have been created by Tatu Saloranta.

For additional credits (generally to people who reported problems) see CREDITS file.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of

the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works

that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A

PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.332 wire-multiplatform-runtime- (experimental) 3.6.0

1.332.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.333 java-architecture-for-xml-binding 2.3.3

1.333.1 Available under license :

(See license.txt for the actual license terms)

Copyright 2001-@@YEAR@@ Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California, 94303, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in this product. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and other countries. This product is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third party software, including font technology, is copyrighted and licensed from Sun suppliers. Sun, the Sun logo, and Sun Microsystems are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Federal Acquisitions: Commercial Software - Government Users Subject to Standard License Terms and Conditions.

Copyright 2001--@@YEAR@@ Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 tats-Unis. Tous droits rservs. Distribue par des licences qui en restreignent l'utilisation. Sun Microsystems, Inc. a les droits de propriit intellectuels relatants la technologie incorpore dans ce produit. En particulier, et sans la limitation, ces droits de propriit intellectuels peuvent inclure un ou plus des brevets amricains numrs <http://www.sun.com/patents> et un ou les brevets plus supplmentaires ou les applications de brevet en attente dans les Etats Unis et les autres pays. Ce produit ou document est protg par un copyright et distribu avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la dcompilation. Aucune partie de ce produit ou document ne peut tre reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation pralable et crite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel dtenu par des tiers, et qui comprend la technologie relative aux polices de caractres, est protg par un copyright et licenci par des fournisseurs de Sun. Sun, le logo Sun, Sun Microsystems et sont des marques de fabrique ou des marques d?pos?es de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Ce produit inclut le logiciel dvelopp par la base de Apache Software Foundation (<http://www.apache.org/>). L'accord du gouvernement des tats Unis est requis avant l'exportation du produit.

/*

* The Apache Software License, Version 1.1

*

*

* Copyright (c) 1999-2004 The Apache Software Foundation. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

* "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)."

- * Alternately, this acknowledgment may appear in the software itself,
- * if and wherever such third-party acknowledgments normally appear.

*

- * 4. The names "Xerces" and "Apache Software Foundation" must
- * not be used to endorse or promote products derived from this
- * software without prior written permission. For written
- * permission, please contact apache@apache.org.

*

- * 5. Products derived from this software may not be called "Apache",
- * nor may "Apache" appear in their name, without prior written
- * permission of the Apache Software Foundation.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*

- * This software consists of voluntary contributions made by many
- * individuals on behalf of the Apache Software Foundation and was
- * originally based on software copyright (c) 1999, International
- * Business Machines, Inc., <http://www.apache.org>. For more
- * information on the Apache Software Foundation, please see
- * [<http://www.apache.org/>](http://www.apache.org/).

*/

THIS LICENSE IS INTENDED TO BE USED FOR DEBUGGING THE INSTALLER.

Amendment I

Congress shall make no law respecting an establishment of religion, or prohibiting the free exercise thereof; or abridging the freedom of speech, or of the press; or the right of the people peaceably to assemble, and to petition the government for a redress of grievances.

Amendment II

A well regulated militia, being necessary to the security of a free state, the right of the people to keep and bear arms, shall not be infringed.

Amendment III

No soldier shall, in time of peace be quartered in any house, without the consent of the owner, nor in time of war, but in a manner to be prescribed by law.

Amendment IV

The right of the people to be secure in their persons, houses, papers, and effects, against unreasonable searches and seizures, shall not be violated, and no warrants shall issue, but upon probable cause, supported by oath or affirmation, and particularly describing the place to be searched, and the persons or things to be seized.

Amendment V

No person shall be held to answer for a capital, or otherwise infamous crime, unless on a presentment or indictment of a grand jury, except in cases arising in the land or naval forces, or in the militia, when in actual service in time of war or public danger; nor shall any person be subject for the same offense to be twice put in jeopardy of life or limb; nor shall be compelled in any criminal case to be a witness against himself, nor be deprived of life, liberty, or property, without due process of law; nor shall private property be taken for public use, without just compensation.

Amendment VI

In all criminal prosecutions, the accused shall enjoy the right to a speedy and public trial, by an impartial jury of the state and district wherein the crime shall have been committed, which district shall have been previously ascertained by law, and to be informed of the nature and cause of the accusation; to be confronted with the witnesses against him; to have compulsory process for obtaining witnesses in his favor, and to have the assistance of counsel for his defense.

Amendment VII

In suits at common law, where the value in controversy shall exceed twenty dollars, the right of trial by jury shall be preserved, and no fact tried by a jury, shall be otherwise reexamined in any court of the United States, than according to the rules of the common law.

Amendment VIII

Excessive bail shall not be required, nor excessive fines imposed, nor cruel and unusual punishments inflicted.

Amendment IX

The enumeration in the Constitution, of certain rights, shall not be construed to deny or disparage others retained by the people.

Amendment X

The powers not delegated to the United States by the Constitution, nor prohibited by it to the states, are reserved to the states respectively, or to the people.

(See license.txt for the actual license terms)

Copyright 2001-@@YEAR@@ Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California, 94303, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in this product. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and other countries. This product is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third party software, including font technology, is copyrighted and licensed from Sun suppliers. Sun, the Sun logo, and Sun Microsystems are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Federal Acquisitions: Commercial Software - Government Users Subject to Standard License Terms and Conditions.

Copyright 2001-@@YEAR@@ Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 tats-Unis. Tous droits rservs. Distribue par des licences qui en restreignent l'utilisation. Sun Microsystems, Inc. a les droits de propriit intellectuels relatants la technologie incorpore dans ce produit. En particulier, et sans la limitation, ces droits de propriit intellectuels peuvent inclure un ou plus des brevets amricains numrs <http://www.sun.com/patents> et un ou les brevets plus supplmentaires ou les applications de brevet en attente dans les Etats Unis et les autres pays. Ce produit ou document est protg par un copyright et distribu avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la dcompilation. Aucune partie de ce produit ou document ne peut tre reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation pralable et crite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel dtenu par des tiers, et qui comprend la technologie relative aux polices de caractres, est protg par un copyright et licenci par des fournisseurs de Sun. Sun, le logo Sun, Sun Microsystems et sont des marques de fabrique ou des marques d'pos'es de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Ce produit inclut le logiciel dvelopp par la base de Apache Software Foundation (<http://www.apache.org/>). L'accord du gouvernement des tats Unis est requis avant l'exportation du produit.

/* =====
* The Apache Software License, Version 1.1
*
* Copyright (c) 2001-2003 The Apache Software Foundation. All rights
* reserved.
*

* Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution,
 * if any, must include the following acknowledgment:
 * "This product includes software developed by the
 * Apache Software Foundation (<http://www.apache.org/>)."
 * Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *
 * 4. The names "Apache" and "Apache Software Foundation" must
 * not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 *
 * 5. Products derived from this software may not be called "Apache",
 * nor may "Apache" appear in their name, without prior written
 * permission of the Apache Software Foundation.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 *
 * =====
 *
 * This software consists of voluntary contributions made by many
 * individuals on behalf of the Apache Software Foundation. For more
 * information on the Apache Software Foundation, please see
 * <http://www.apache.org/>.
 */

```
/* =====
* The Apache Software License, Version 1.1
*
* Copyright (c) 2000 The Apache Software Foundation. All rights
* reserved.
*
* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:
*
* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.
*
* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.
*
* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (http://www.apache.org/)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.
*
* 4. The names "Apache" and "Apache Software Foundation" must
* not be used to endorse or promote products derived from this
* software without prior written permission. For written
* permission, please contact apache@apache.org.
*
* 5. Products derived from this software may not be called "Apache",
* nor may "Apache" appear in their name, without prior written
* permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
* =====
*
```

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <<http://www.apache.org/>>.
*
* Portions of this software are based upon public domain software
* originally written at the National Center for Supercomputing Applications,
* University of Illinois, Urbana-Champaign.
*/

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed

with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate

comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.

1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. "Executable" means the Covered Software in any form other than Source Code.

1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.

1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. "License" means this document.

1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. "Modifications" means the Source Code and Executable form of any of the following:

- A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
- B. Any new file that contains any part of the Original Software or previous Modification; or
- C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making,

using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to

grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTYS NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The

application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

Copyright (c) 2001, Sun Microsystems, Inc.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of Sun Microsystems, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDERS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)

ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This license came from: <http://www.megginson.com/SAX/copying.html>

However please note future versions of SAX may be covered under <http://saxproject.org/?selected=pd>

This page is now out of date -- see the new SAX site at <http://www.saxproject.org/> for more up-to-date releases and other information. Please change your bookmarks.

SAX2 is Free!

I hereby abandon any property rights to SAX 2.0 (the Simple API for XML), and release all of the SAX 2.0 source code, compiled code, and documentation contained in this distribution into the Public Domain. SAX comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson, david@megginson.com
2000-05-05

This license came from:

<http://www.w3.org/Consortium/Legal/copyright-software-19980720>

W3C SOFTWARE NOTICE AND LICENSE

Copyright 1994-2001 World

Wide Web Consortium, (<http://www.w3.org/>) World

Wide Web Consortium

(<http://www.lcs.mit.edu/>) Massachusetts Institute of

Technology, (<http://www.inria.fr/>) Institut National de

Recherche en Informatique et en Automatique, (<http://www.keio.ac.jp/>) Keio University).

All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

This W3C work (including software, documents, or other related items) is being provided by the copyright holders under the following license. By obtaining, using and/or copying this work, you (the licensee) agree that you have read, understood, and will comply with the following terms and conditions:

Permission to use, copy, modify, and distribute this software and its documentation, with or without modification, for any purpose and without fee or royalty is hereby granted, provided that you include the following on ALL copies of the software and documentation or portions thereof, including modifications, that you make:

The full text of this NOTICE in a location viewable to users of the redistributed or derivative work.

Any pre-existing intellectual property disclaimers, notices, or terms and conditions. If none exist, a short notice of the following form (hypertext is preferred, text is permitted) should be used within the body of any redistributed or derivative code:
"Copyright [\$date-of-software] World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/>"

Notice of any changes or modifications to the W3C files, including the date changes were made. (We recommend you provide URIs to the location from which the code is derived.)

THIS SOFTWARE AND DOCUMENTATION IS PROVIDED "AS IS," AND COPYRIGHT HOLDERS MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR THAT THE USE OF THE SOFTWARE OR DOCUMENTATION WILL NOT INFRINGE ANY THIRD PARTY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS. COPYRIGHT HOLDERS WILL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF ANY USE OF THE SOFTWARE OR DOCUMENTATION.

The name and trademarks of copyright holders may NOT be used in advertising or publicity pertaining to the software without specific, written prior permission. Title to copyright in this software and any associated documentation will at all times remain with copyright holders.

This formulation of W3C's notice and license became active on August 14 1998 so as to improve compatibility with GPL. This version ensures that W3C software licensing terms are no more restrictive than GPL and consequently W3C software may be distributed in GPL packages. See the older formulation for the policy prior to this date. Please see our Copyright FAQ for common questions about using materials from our site, including specific terms and conditions for packages like libwww, Amaya, and Jigsaw. Other questions about this notice can be directed to site-policy@w3.org.

webmaster
Sun Microsystems, Inc.
Binary Code License Agreement

READ THE TERMS OF THIS AGREEMENT AND ANY PROVIDED SUPPLEMENTAL LICENSE TERMS (COLLECTIVELY "AGREEMENT") CAREFULLY BEFORE OPENING THE SOFTWARE MEDIA PACKAGE. BY OPENING THE SOFTWARE MEDIA PACKAGE, YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ACCESSING THE SOFTWARE ELECTRONICALLY, INDICATE YOUR ACCEPTANCE OF THESE TERMS BY SELECTING THE "ACCEPT" BUTTON AT THE END OF THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL THESE TERMS, PROMPTLY RETURN THE UNUSED SOFTWARE TO YOUR PLACE OF PURCHASE FOR A REFUND OR, IF THE SOFTWARE IS ACCESSED ELECTRONICALLY, SELECT THE "DECLINE" BUTTON AT THE END OF THIS AGREEMENT.

1. LICENSE TO USE. Sun grants you a non-exclusive and non-transferable license for the internal use only of the accompanying software and documentation and any error corrections provided by Sun (collectively "Software"), by the number of users and the class of computer hardware for which the corresponding fee has been paid.

2. RESTRICTIONS. Software is confidential and copyrighted. Title to Software and all associated intellectual property rights is retained by Sun and/or its licensors. Except as specifically authorized in any Supplemental License Terms, you may not make copies of Software, other than a single copy of Software for archival purposes. Unless enforcement is prohibited by applicable law, you may not modify, decompile, or reverse engineer Software. You acknowledge that Software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility. Sun disclaims any express or implied warranty of fitness for such uses. No right, title or interest in or to any trademark, service mark, logo or trade name of Sun or its licensors is granted under this Agreement.

3. LIMITED WARRANTY. Sun warrants to you that for a period of ninety (90) days from the date of purchase, as evidenced by a copy of the receipt, the media on which Software is furnished (if any) will be free of defects in materials and workmanship under normal use. Except for the foregoing, Software is provided "AS IS". Your exclusive remedy and Sun's entire liability under this limited warranty will be at Sun's option to replace Software media or refund the fee paid for Software.

4. DISCLAIMER OF WARRANTY. UNLESS SPECIFIED IN THIS AGREEMENT, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED

WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT THESE DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

5. LIMITATION OF LIABILITY. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED TO THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

In no event will Sun's liability to you, whether in contract, tort (including negligence), or otherwise, exceed the amount paid by you for Software under this Agreement.

The foregoing limitations will apply even if the above stated warranty fails of its essential purpose.

6. Termination. This Agreement is effective until terminated. You may terminate this Agreement at any time by destroying all copies of Software. This Agreement will terminate immediately without notice from Sun if you fail to comply with any provision of this Agreement. Upon Termination, you must destroy all copies of Software.

7. Export Regulations. All Software and technical data delivered under this Agreement are subject to US export control laws and may be subject to export or import regulations in other countries. You agree to comply strictly with all such laws and regulations and acknowledge that you have the responsibility to obtain such licenses to export, re-export, or import as may be required after delivery to you.

8. U.S. Government Restricted Rights. If Software is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in Software and accompanying documentation will be only as set forth in this Agreement; this is in accordance with 48 CFR 227.7201 through 227.7202-4 (for Department of Defense (DOD) acquisitions) and with 48 CFR 2.101 and 12.212 (for non-DOD acquisitions).

9. Governing Law. Any action related to this Agreement will be governed by California law and controlling U.S. federal law. No choice of law rules of any jurisdiction will apply.

10. Severability. If any provision of this Agreement is held to be unenforceable, this Agreement will remain in effect with the provision omitted, unless omission would frustrate the intent of the parties, in which case this Agreement will immediately terminate.

11. Integration. This Agreement is the entire agreement between you and Sun relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification of this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

SUN XML INSTANCE GENERATOR, VERSION 1.0 SUPPLEMENTAL LICENSE TERMS

These supplemental license terms ("Supplemental Terms") add to or modify the terms of the Binary Code License Agreement (collectively, the "Agreement"). Capitalized terms not defined in these Supplemental Terms shall have the same meanings ascribed to them in the Agreement. These Supplemental Terms shall supersede any inconsistent or conflicting terms in the Agreement, or in any license contained within the Software.

1. Software Internal Use and Development License Grant. Subject to the terms and conditions of this Agreement, including, but not limited to Section 4 (Java(TM) Technology Restrictions) of these Supplemental Terms, Sun grants to you, a non-exclusive, non-transferable, royalty-free and limited license to reproduce, modify, and create derivative works of the Software for the sole purpose of adding value and improving the Software for the development of applications ("Programs").
2. License to Distribute Software. Subject to the terms and conditions of this Agreement, including, but not limited to Section 4 (Java (TM) Technology Restrictions) of these Supplemental Terms, Sun grants you a non-exclusive, non-transferable, limited license to reproduce and distribute the Software modified by you as permitted in Section 1 of these Supplemental Terms ("Modified Software")

in source or binary code form, provided that (i) you distribute the Modified Software only bundled as part of, and for the sole purpose of running, your Programs, (ii) the Modified Software adds value and improveS the function of the Software, (iv) you do not remove or alter any proprietary legends or notices contained in the Software, (v) you only distribute the Modified Software subject to a license agreement that protects Sun's interests consistent with the terms contained in this Agreement, and (vi) you agree to defend and indemnify Sun and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorneys' fees) incurred in connection with any claim, lawsuit or action by any third party that arises or results from the use or distribution of any and all Programs and/or Modified Software.

3. Experimental Software. You acknowledge that the Software is experimental and may contain errors, defects, or deficiencies which cannot or will not be corrected by Sun. You shall have the sole responsibility to protect adequately and backup your data and/or equipment used in connection with the Software. You shall not claim against Sun for lost data, re-run time, inaccurate output, work delays or lost profits resulting from your use of the Licensed Software.

4. Java Technology Restrictions. You may not modify the Java Platform Interface ("JPI", identified as classes contained within the "java" package or any subpackages of the "java" package), by creating additional classes within the JPI or otherwise causing the addition to or modification of the classes in the JPI. In the event that you create an additional class and associated API(s) which (i) extends the functionality of the Java platform, and (ii) is exposed to third party software developers for the purpose of developing additional software which invokes such additional API, you must promptly publish broadly an accurate specification for such API for free use by all developers. You may not create, or authorize your licensees to create, additional classes, interfaces, or subpackages that are in any way identified as "java", "javax", "sun" or similar convention as specified by Sun in any naming convention designation.

5. Trademarks and Logos. You acknowledge and agree as between you and Sun that Sun owns the SUN, SOLARIS, JAVA, JINI, FORTE, and iPLANET trademarks and all SUN, SOLARIS, JAVA, JINI, FORTE, and iPLANET-related trademarks, service marks, logos and other brand designations ("Sun Marks"), and

you agree to comply with the Sun Trademark and Logo Usage Requirements currently located at <http://www.sun.com/policies/trademarks>. Any use you make of the Sun Marks inures to Sun's benefit.

6. Termination for Infringement. Either party may terminate this Agreement immediately should any Software become, or in either party's opinion be likely to become, the subject of a claim of infringement of any intellectual property right.

For inquiries please contact:

Sun Microsystems, Inc.

901 San Antonio Road, Palo Alto, California 94303

(LFI#100313/Form ID#011801)

Copyright (c) 2003, Kohsuke Kawaguchi

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright (c) 2003 Sun Microsystems, Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

-Redistribution of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

-Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN MIDROSYSTEMS, INC. ("SUN") AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES SUFFERED BY LICENSEE AS A RESULT OF USING, MODIFYING OR DISTRIBUTING THIS SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE THIS SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You acknowledge that this software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Sun Microsystems, Inc.
Binary Code License Agreement

READ THE TERMS OF THIS AGREEMENT AND ANY PROVIDED SUPPLEMENTAL LICENSE TERMS (COLLECTIVELY "AGREEMENT") CAREFULLY BEFORE OPENING THE SOFTWARE MEDIA PACKAGE. BY OPENING THE SOFTWARE MEDIA PACKAGE, YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ACCESSING THE SOFTWARE ELECTRONICALLY, INDICATE YOUR ACCEPTANCE OF THESE TERMS BY SELECTING THE "ACCEPT" BUTTON AT THE END OF THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL THESE TERMS, PROMPTLY RETURN THE UNUSED SOFTWARE TO YOUR PLACE OF PURCHASE FOR A REFUND OR, IF THE SOFTWARE IS ACCESSED ELECTRONICALLY, SELECT THE "DECLINE" BUTTON AT THE END OF THIS AGREEMENT.

1. LICENSE TO USE. Sun grants you a non-exclusive and non-transferable license for the internal use only of the accompanying software and documentation and any error corrections provided by Sun (collectively "Software"), by the number of users and the class of computer hardware for which the corresponding fee has been paid.

2. **RESTRICTIONS.** Software is confidential and copyrighted. Title to Software and all associated intellectual property rights is retained by Sun and/or its licensors. Except as specifically authorized in any Supplemental License Terms, you may not make copies of Software, other than a single copy of Software for archival purposes. Unless enforcement is prohibited by applicable law, you may not modify, decompile, or reverse engineer Software. You acknowledge that Software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility. Sun disclaims any express or implied warranty of fitness for such uses. No right, title or interest in or to any trademark, service mark, logo or trade name of Sun or its licensors is granted under this Agreement.

3. **LIMITED WARRANTY.** Sun warrants to you that for a period of ninety (90) days from the date of purchase, as evidenced by a copy of the receipt, the media on which Software is furnished (if any) will be free of defects in materials and workmanship under normal use. Except for the foregoing, Software is provided "AS IS". Your exclusive remedy and Sun's entire liability under this limited warranty will be at Sun's option to replace Software media or refund the fee paid for Software.

4. **DISCLAIMER OF WARRANTY.** UNLESS SPECIFIED IN THIS AGREEMENT, ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT ARE DISCLAIMED, EXCEPT TO THE EXTENT THAT THESE DISCLAIMERS ARE HELD TO BE LEGALLY INVALID.

5. **LIMITATION OF LIABILITY.** TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF OR RELATED TO THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. In no event will Sun's liability to you, whether in contract, tort (including negligence), or otherwise, exceed the amount paid by you for Software under this Agreement. The foregoing limitations will apply even if the above stated warranty fails of its essential purpose.

6. **Termination.** This Agreement is effective until terminated. You may terminate this Agreement at any time by destroying all copies of Software. This Agreement will terminate immediately without notice from Sun if you fail to comply with any provision of this Agreement. Upon Termination, you must destroy all copies of Software.

7. **Export Regulations.** All Software and technical data delivered under this Agreement are subject to US export control laws and may be subject to export or import regulations in other countries. You agree to comply strictly with all such laws and regulations and acknowledge that you have the

responsibility to obtain such licenses to export, re-export, or import as may be required after delivery to you.

8. U.S. Government Restricted Rights. If Software is being acquired by or on behalf of the U.S. Government or by a U.S. Government prime contractor or subcontractor (at any tier), then the Government's rights in Software and accompanying documentation will be only as set forth in this Agreement; this is in accordance with 48 CFR 227.7201 through 227.7202-4 (for Department of Defense (DOD) acquisitions) and with 48 CFR 2.101 and 12.212 (for non-DOD acquisitions).

9. Governing Law. Any action related to this Agreement will be governed by California law and controlling U.S. federal law. No choice of law rules of any jurisdiction will apply.

10. Severability. If any provision of this Agreement is held to be unenforceable, this Agreement will remain in effect with the provision omitted, unless omission would frustrate the intent of the parties, in which case this Agreement will immediately terminate.

11. Integration. This Agreement is the entire agreement between you and Sun relating to its subject matter. It supersedes all prior or contemporaneous oral or written communications, proposals, representations and warranties and prevails over any conflicting or additional terms of any quote, order, acknowledgment, or other communication between the parties relating to its subject matter during the term of this Agreement. No modification of this Agreement will be binding, unless in writing and signed by an authorized representative of each party.

JAVA OPTIONAL PACKAGE

JAVABEANS(TM) ACTIVATION FRAMEWORK, VERSION 1.0.2 SUPPLEMENTAL LICENSE TERMS

These supplemental license terms ("Supplemental Terms") add to or modify the terms of the Binary Code License Agreement (collectively, the "Agreement"). Capitalized terms not defined in these Supplemental Terms shall have the same meanings ascribed to them in the Agreement. These Supplemental Terms shall supersede any inconsistent or conflicting terms in the Agreement, or in any license contained within the Software.

1. Software Internal Use and Development License Grant. Subject to the terms and conditions of this Agreement, including, but not limited to Section 3 (Java(TM) Technology Restrictions) of these Supplemental Terms, Sun grants you a non-exclusive, non-transferable, limited license to reproduce internally and use internally the binary form of the Software, complete and unmodified, for the sole purpose of designing, developing and testing your Java applets and applications ("Programs").

2. License to Distribute Software. In addition to the license granted in Section 1 (Software Internal Use and Development License Grant) of these Supplemental Terms, subject to the terms and conditions of this Agreement, including but not limited to, Section 3 (Java Technology Restrictions) of these Supplemental Terms, Sun grants you a non-exclusive, non-transferable, limited license to reproduce and distribute the Software in binary code form only, provided that you (i) distribute the Software complete and unmodified and only bundled as part of your Programs, (ii) do not distribute additional software intended to replace any component(s) of the Software, (iii) do not remove or alter any proprietary legends or notices contained in the Software, (iv) only distribute the Software subject to a license agreement that protects Sun's interests consistent with the terms contained in this Agreement, and (v) agree to defend and indemnify Sun and its licensors from and against any damages, costs, liabilities, settlement amounts and/or expenses (including attorneys' fees) incurred in connection with any claim, lawsuit or action by any third party that arises or results from the use or distribution of any and all Programs and/or Software.

3. Java Technology Restrictions. You may not modify the Java Platform Interface ("JPI", identified as classes contained within the "java" package or any subpackages of the "java" package), by creating additional classes within the JPI or otherwise causing the addition to or modification of the classes in the JPI. In the event that you create an additional class and associated API(s) which (i) extends the functionality of the Java platform, and (ii) is exposed to third party software developers for the purpose of developing additional software which invokes such additional API, you must promptly publish broadly an accurate specification for such API for free use by all developers. You may not create, or authorize your licensees to create additional classes, interfaces, or subpackages that are in any way identified as "java", "javax", "sun" or similar convention as specified by Sun in any naming convention designation.

4. No Support. Sun is under no obligation to support the Software or to provide you with updates or error corrections. You acknowledge that the Software may have defects or deficiencies which cannot or will not be corrected by Sun.

5. Trademarks and Logos. You acknowledge and agree as between you and Sun that Sun owns the SUN, SOLARIS, JAVA, JINI, FORTE, and iPLANET trademarks and all SUN, SOLARIS, JAVA, JINI, FORTE, and iPLANET-related trademarks, service marks, logos and other brand designations ("Sun Marks"), and you agree to comply with the Sun Trademark and Logo Usage Requirements currently located at <http://www.sun.com/policies/trademarks>. Any use you make of the Sun Marks inures to Sun's benefit.

6. Source Code. Software may contain source code that is provided solely for reference purposes pursuant to the terms of this Agreement. Source code may

not be redistributed unless expressly provided for in this Agreement.

7. Termination for Infringement. Either party may terminate this Agreement immediately should any Software become, or in either party's opinion be likely to become, the subject of a claim of infringement of any intellectual property right.

For inquiries please contact: Sun Microsystems, Inc. 901 San Antonio Road,
Palo Alto, California 94303
(LFI#115020/Form ID#011801)

/*

* The Apache Software License, Version 1.1

*

*

* Copyright (c) 1999-2002 The Apache Software Foundation. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

* "This product includes software developed by the
* Apache Software Foundation (<http://www.apache.org/>)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.

*

* 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

*

* 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.

*

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE

* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999, International
* Business Machines, Inc., <http://www.ibm.com>. For more
* information on the Apache Software Foundation, please see
* <<http://www.apache.org/>>.

*/

Copyright (c) 2004 Kohsuke Kawaguchi

Permission is hereby granted, free of charge, to any person
obtaining a copy of this software and associated documentation
files (the "Software"), to deal in the Software without
restriction, including without limitation the rights to use,
copy, modify, merge, publish, distribute, sublicense, and/or
sell copies of the Software, and to permit persons to whom
the Software is furnished to do so, subject to the following
conditions:

The above copyright notice and this permission notice shall
be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY
KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE
WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR
PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS
OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR
OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR
OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or
contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original

Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity

exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such

Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not

remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software

(meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this

license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and

modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an

appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not

distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the

author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify

it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show
w'. This is free software, and you are welcome to redistribute it under
certain conditions; type `show c' for details.
```

The hypothetical commands ``show w'` and ``show c'` should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than ``show w'` and ``show c'`; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by James
Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version. Copyright (c) 2001-@@YEAR@@ Sun Microsystems, Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES OR LIABILITIES SUFFERED BY LICENSEE AS A RESULT OF OR RELATING TO USE, MODIFICATION OR DISTRIBUTION OF THE SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL

SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You acknowledge that Software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

Copyright (c) 2003, Kohsuke Kawaguchi

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2001-2005 (C) MetaStuff, Ltd. All Rights Reserved.

Redistribution and use of this software and associated documentation ("Software"), with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain copyright

statements and notices. Redistributions must also contain a copy of this document.

2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "DOM4J" must not be used to endorse or promote products derived from this Software without prior written permission of MetaStuff, Ltd. For written permission, please contact dom4j-info@metastuff.com.
4. Products derived from this Software may not be called "DOM4J" nor may "DOM4J" appear in their names without prior written permission of MetaStuff, Ltd. DOM4J is a registered trademark of MetaStuff, Ltd.
5. Due credit should be given to the DOM4J Project - <http://www.dom4j.org>

THIS SOFTWARE IS PROVIDED BY METASTUFF, LTD. AND CONTRIBUTORS ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL METASTUFF, LTD. OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Copyright 2001-@@YEAR@@ Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto, California, 94303, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to technology embodied in this product. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.sun.com/patents> and one or more additional patents or pending patent applications in the U.S. and other countries. This product is distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any. Third party software, including font technology, is copyrighted and licensed from Sun suppliers. Sun, the Sun logo, and Sun Microsystems are trademarks or registered trademarks of Sun Microsystems, Inc. in the

U.S. and other countries. This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Federal Acquisitions: Commercial Software - Government Users Subject to Standard License Terms and Conditions.

Copyright 2001-@@YEAR@@ Sun Microsystems, Inc., 901 San Antonio Road, Palo Alto, Californie 94303 tats-Unis. Tous droits rservs. Distribue par des licences qui en restreignent l'utilisation. Sun Microsystems, Inc. a les droits de propriit intellectuels relatants la technologie incorpore dans ce produit. En particulier, et sans la limitation, ces droits de propriit intellectuels peuvent inclure un ou plus des brevets amricains numrs <http://www.sun.com/patents> et un ou les brevets plus supplmentaires ou les applications de brevet en attente dans les Etats Unis et les autres pays. Ce produit ou document est protg par un copyright et distribu avec des licences qui en restreignent l'utilisation, la copie, la distribution, et la dcompilation. Aucune partie de ce produit ou document ne peut tre reproduite sous aucune forme, par quelque moyen que ce soit, sans l'autorisation pralable et crite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel dtenu par des tiers, et qui comprend la technologie relative aux polices de caractres, est protg par un copyright et licenci par des fournisseurs de Sun. Sun, le logo Sun, Sun Microsystems et sont des marques de fabrique ou des marques d'pos?es de Sun Microsystems, Inc. aux Etats-Unis et dans d'autres pays. Ce produit inclut le logiciel dvelopp par la base de Apache Software Foundation (<http://www.apache.org/>). L'accord du gouvernement des tats Unis est requis avant l'exportation du produit.
COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the

Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY.S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction.s conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys. fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its

contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER

PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

Copyright (c) 2001-2005 Sun Microsystems, Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES OR LIABILITIES

SUFFERED BY LICENSEE AS A RESULT OF OR RELATING TO USE, MODIFICATION OR DISTRIBUTION OF THE SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

You acknowledge that Software is not designed, licensed or intended for use in the design, construction, operation or maintenance of any nuclear facility.

```
=====
== NOTICE file corresponding to the section 4 d of           ==
== the Apache License, Version 2.0,                          ==
== in this case for the Apache Ant distribution.             ==
=====
```

This product includes software developed by
The Apache Software Foundation (<http://www.apache.org/>).

This product includes also software developed by :
- the W3C consortium (<http://www.w3c.org/>) ,
- the SAX project (<http://www.saxproject.org/>)

Please read the different LICENSE files present in the root directory of this distribution.

```
/*
 * The Apache Software License, Version 1.1
 *
 * Copyright (c) 2001-2003 Ant-Contrib project. All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 * notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 * notice, this list of conditions and the following disclaimer in
 * the documentation and/or other materials provided with the
 * distribution.
 *
 * 3. The end-user documentation included with the redistribution, if
 * any, must include the following acknowledgement:
 * "This product includes software developed by the
 * Ant-Contrib project (http://sourceforge.net/projects/ant-contrib/)."
```

* if and wherever such third-party acknowledgements normally appear.
 *
 * 4. The name Ant-Contrib must not be used to endorse or promote products
 * derived from this software without prior written permission. For
 * written permission, please contact
 * ant-contrib-developers@lists.sourceforge.net.
 *
 * 5. Products derived from this software may not be called "Ant-Contrib"
 * nor may "Ant-Contrib" appear in their names without prior written
 * permission of the Ant-Contrib project.
 *
 * THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE ANT-CONTRIB PROJECT OR ITS
 * CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 */
 /*
 * \$Id: license.txt,v 1.2 2006/04/01 06:01:50 jeffsuttor Exp \$
 * %W% %E%
 */

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0 1.

Definitions.

- 1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.
- 1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
- 1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
- 1.4. Executable means the Covered Software in any form other than Source Code.

- 1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.
- 1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
- 1.7. License means this document.
- 1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
- 1.9. Modifications means the Source Code and Executable form of any of the following: A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications; B. Any new file that contains any part of the Original Software or previous Modification; or C. Any new file that is contributed or otherwise made available under the terms of this License.
- 1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.
- 1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.
- 1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.
- 1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

- 2.1. The Initial Developer Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual

property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof);

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License;

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant. Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the

Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code. Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications. The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices. You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms. You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every

Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions. You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works. You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions. Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions. You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions. When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY. COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY. UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS. The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS. This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale

of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS. As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed

as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. **Limitation of Liability.** In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. **Accepting Warranty or Additional Liability.** While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this

License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Copyright (c) 2002-@@YEAR@@ Sun Microsystems, Inc. All Rights Reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistribution in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the name of Sun Microsystems, Inc. or the names of contributors may be used to endorse or promote products derived from

this software without specific prior written permission.

This software is provided "AS IS," without a warranty of any kind. ALL EXPRESS OR IMPLIED CONDITIONS, REPRESENTATIONS AND WARRANTIES, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT, ARE HEREBY EXCLUDED. SUN AND ITS LICENSORS SHALL NOT BE LIABLE FOR ANY DAMAGES OR LIABILITIES SUFFERED BY LICENSEE AS A RESULT OF OR RELATING TO USE, MODIFICATION OR DISTRIBUTION OF THE SOFTWARE OR ITS DERIVATIVES. IN NO EVENT WILL SUN OR ITS LICENSORS BE LIABLE FOR ANY LOST REVENUE, PROFIT OR DATA, OR FOR DIRECT, INDIRECT, SPECIAL, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF THE USE OF OR INABILITY TO USE SOFTWARE, EVEN IF SUN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The CyberNeko Software License, Version 1.0

(C) Copyright 2002-2005, Andy Clark. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
"This product includes software developed by Andy Clark."
Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
4. The names "CyberNeko" and "NekoHTML" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact andyc@cyberneko.net.
5. Products derived from this software may not be called "CyberNeko", nor may "CyberNeko" appear in their name, without prior written permission of the author.

THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES

OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR OTHER CONTRIBUTORS
BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY,
OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT
OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR
BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY,
WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE
OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE,
EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

=====
This license is based on the Apache Software License, version 1.1.
Copyright 2001 Sun Microsystems, Inc. 901 San Antonio Road, Palo Alto,
California, 94303, U.S.A. All rights reserved.

Sun Microsystems, Inc. has intellectual property rights relating to
technology embodied in this product. In particular, and without
limitation, these intellectual property rights may include one or more
of the U.S. patents listed at <http://www.sun.com/patents> and one or
more additional patents or pending patent applications in the U.S. and
other countries. This product is distributed under licenses
restricting its use, copying, distribution, and decompilation. No part
of this product may be reproduced in any form by any means without
prior written authorization of Sun and its licensors, if any. Third
party software, including font technology, is copyrighted and licensed
from Sun suppliers. Sun, the Sun logo, and Sun Microsystems are
trademarks or registered trademarks of Sun Microsystems, Inc. in the
U.S. and other countries. This product includes software developed by
the Apache Software Foundation (<http://www.apache.org/>). Federal
Acquisitions: Commercial Software - Government Users Subject to
Standard License Terms and Conditions.

Copyright 2001 Sun Microsystems, Inc., 901 San Antonio Road, Palo
Alto, Californie 94303 tats-Unis. Tous droits rservs. Distribue
par des licences qui en restreignent l'utilisation. Sun Microsystems,
Inc. a les droits de proprit intellectuels relatants la
technologie incorpore dans ce produit. En particulier, et sans la
limitation, ces droits de proprit intellectuels peuvent inclure un
ou plus des brevets amricains numrs <http://www.sun.com/patents>
et un ou les brevets plus supplmentaires ou les applications de
brevet en attente dans les Etats Unis et les autres pays. Ce produit
ou document est protg par un copyright et distribu avec des
licences qui en restreignent l'utilisation, la copie, la distribution,
et la dcompilation. Aucune partie de ce produit ou document ne peut
tre reproduite sous aucune forme, par quelque moyen que ce soit, sans

l'autorisation préalable et écrite de Sun et de ses bailleurs de licence, s'il y en a. Le logiciel détenu par des tiers, et qui comprend la technologie relative aux polices de caractères, est protégé par un copyright et licencié par des fournisseurs de Sun. Sun, le logo Sun, Sun Microsystems et sont des marques de fabrique ou des marques déposées de Sun Microsystems, Inc. aux États-Unis et dans d'autres pays. Ce produit inclut le logiciel développé par la base de Apache Software Foundation (<http://www.apache.org/>). L'accord du gouvernement des États-Unis est requis avant l'exportation du produit.

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.1

1. Definitions.

- 1.1. "Contributor" means each individual or entity that creates or contributes to the creation of Modifications.
- 1.2. "Contributor Version" means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.
- 1.3. "Covered Software" means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.
- 1.4. "Executable" means the Covered Software in any form other than Source Code.
- 1.5. "Initial Developer" means the individual or entity that first makes Original Software available under this License.
- 1.6. "Larger Work" means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.
- 1.7. "License" means this document.
- 1.8. "Licensable" means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.
- 1.9. "Modifications" means the Source Code and Executable form of any of the following:
 - A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;
 - B. Any new file that contains any part of the Original Software or previous Modification; or
 - C. Any new file that is contributed or otherwise made available under the terms of this License.
- 1.10. "Original Software" means the Source Code and Executable form of computer software code that is originally released under this License.
- 1.11. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. "Source Code" means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or

otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the

Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Oracle is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN "AS IS" BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABLE, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as "Participant") alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. If You assert a patent infringement claim against Participant alleging that the Participant Software directly or indirectly infringes any patent where such claim is resolved (such as by license or settlement) prior to the initiation of patent infringement litigation, then the reasonable value of the licenses granted by such Participant under Sections 2.1 or 2.2 shall be taken into account in determining the amount or value of any payment or license.

6.4. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a "commercial item," as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of "commercial computer software" (as that term is defined at 48 C.F.R. ? 252.227-7014(a)(1)) and "commercial computer software documentation" as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer

software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the

software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and

disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other

system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989

Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Oracle are subject to the following clarification and special exception to the GPL Version 2, but only where Oracle has expressly included in the particular source file's header the words "Oracle designates this particular file as subject to the "Classpath" exception as provided by Oracle in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

```
/*
 *           Apache License
 *           Version 2.0, January 2004
 *           http://www.apache.org/licenses/
 *
 * TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION
 *
 * 1. Definitions.
 *
 * "License" shall mean the terms and conditions for use, reproduction,
 * and distribution as defined by Sections 1 through 9 of this document.
 *
 * "Licensor" shall mean the copyright owner or entity authorized by
 * the copyright owner that is granting the License.
 *
 * "Legal Entity" shall mean the union of the acting entity and all
 * other entities that control, are controlled by, or are under common
 * control with that entity. For the purposes of this definition,
 * "control" means (i) the power, direct or indirect, to cause the
 * direction or management of such entity, whether by contract or
 * otherwise, or (ii) ownership of fifty percent (50%) or more of the
 * outstanding shares, or (iii) beneficial ownership of such entity.
 *
 * "You" (or "Your") shall mean an individual or Legal Entity
 * exercising permissions granted by this License.
 *
 * "Source" form shall mean the preferred form for making modifications,
 * including but not limited to software source code, documentation
 * source, and configuration files.
```

*
* "Object" form shall mean any form resulting from mechanical
* transformation or translation of a Source form, including but
* not limited to compiled object code, generated documentation,
* and conversions to other media types.
*
* "Work" shall mean the work of authorship, whether in Source or
* Object form, made available under the License, as indicated by a
* copyright notice that is included in or attached to the work
* (an example is provided in the Appendix below).
*
* "Derivative Works" shall mean any work, whether in Source or Object
* form, that is based on (or derived from) the Work and for which the
* editorial revisions, annotations, elaborations, or other modifications
* represent, as a whole, an original work of authorship. For the purposes
* of this License, Derivative Works shall not include works that remain
* separable from, or merely link (or bind by name) to the interfaces of,
* the Work and Derivative Works thereof.
*
* "Contribution" shall mean any work of authorship, including
* the original version of the Work and any modifications or additions
* to that Work or Derivative Works thereof, that is intentionally
* submitted to Licensor for inclusion in the Work by the copyright owner
* or by an individual or Legal Entity authorized to submit on behalf of
* the copyright owner. For the purposes of this definition, "submitted"
* means any form of electronic, verbal, or written communication sent
* to the Licensor or its representatives, including but not limited to
* communication on electronic mailing lists, source code control systems,
* and issue tracking systems that are managed by, or on behalf of, the
* Licensor for the purpose of discussing and improving the Work, but
* excluding communication that is conspicuously marked or otherwise
* designated in writing by the copyright owner as "Not a Contribution."
*
* "Contributor" shall mean Licensor and any individual or Legal Entity
* on behalf of whom a Contribution has been received by Licensor and
* subsequently incorporated within the Work.
*
* 2. Grant of Copyright License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* copyright license to reproduce, prepare Derivative Works of,
* publicly display, publicly perform, sublicense, and distribute the
* Work and such Derivative Works in Source or Object form.
*
* 3. Grant of Patent License. Subject to the terms and conditions of
* this License, each Contributor hereby grants to You a perpetual,
* worldwide, non-exclusive, no-charge, royalty-free, irrevocable
* (except as stated in this section) patent license to make, have made,

* use, offer to sell, sell, import, and otherwise transfer the Work,
* where such license applies only to those patent claims licensable
* by such Contributor that are necessarily infringed by their
* Contribution(s) alone or by combination of their Contribution(s)
* with the Work to which such Contribution(s) was submitted. If You
* institute patent litigation against any entity (including a
* cross-claim or counterclaim in a lawsuit) alleging that the Work
* or a Contribution incorporated within the Work constitutes direct
* or contributory patent infringement, then any patent licenses
* granted to You under this License for that Work shall terminate
* as of the date such litigation is filed.

*
* 4. Redistribution. You may reproduce and distribute copies of the
* Work or Derivative Works thereof in any medium, with or without
* modifications, and in Source or Object form, provided that You
* meet the following conditions:

* (a) You must give any other recipients of the Work or
* Derivative Works a copy of this License; and

* (b) You must cause any modified files to carry prominent notices
* stating that You changed the files; and

* (c) You must retain, in the Source form of any Derivative Works
* that You distribute, all copyright, patent, trademark, and
* attribution notices from the Source form of the Work,
* excluding those notices that do not pertain to any part of
* the Derivative Works; and

* (d) If the Work includes a "NOTICE" text file as part of its
* distribution, then any Derivative Works that You distribute must
* include a readable copy of the attribution notices contained
* within such NOTICE file, excluding those notices that do not
* pertain to any part of the Derivative Works, in at least one
* of the following places: within a NOTICE text file distributed
* as part of the Derivative Works; within the Source form or
* documentation, if provided along with the Derivative Works; or,
* within a display generated by the Derivative Works, if and
* wherever such third-party notices normally appear. The contents
* of the NOTICE file are for informational purposes only and
* do not modify the License. You may add Your own attribution
* notices within Derivative Works that You distribute, alongside
* or as an addendum to the NOTICE text from the Work, provided
* that such additional attribution notices cannot be construed
* as modifying the License.

* You may add Your own copyright statement to Your modifications and
* may provide additional or different license terms and conditions

* for use, reproduction, or distribution of Your modifications, or
* for any such Derivative Works as a whole, provided Your use,
* reproduction, and distribution of the Work otherwise complies with
* the conditions stated in this License.

*
* 5. Submission of Contributions. Unless You explicitly state otherwise,
* any Contribution intentionally submitted for inclusion in the Work
* by You to the Licensor shall be under the terms and conditions of
* this License, without any additional terms or conditions.
* Notwithstanding the above, nothing herein shall supersede or modify
* the terms of any separate license agreement you may have executed
* with Licensor regarding such Contributions.

*
* 6. Trademarks. This License does not grant permission to use the trade
* names, trademarks, service marks, or product names of the Licensor,
* except as required for reasonable and customary use in describing the
* origin of the Work and reproducing the content of the NOTICE file.

*
* 7. Disclaimer of Warranty. Unless required by applicable law or
* agreed to in writing, Licensor provides the Work (and each
* Contributor provides its Contributions) on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or
* implied, including, without limitation, any warranties or conditions
* of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A
* PARTICULAR PURPOSE. You are solely responsible for determining the
* appropriateness of using or redistributing the Work and assume any
* risks associated with Your exercise of permissions under this License.

*
* 8. Limitation of Liability. In no event and under no legal theory,
* whether in tort (including negligence), contract, or otherwise,
* unless required by applicable law (such as deliberate and grossly
* negligent acts) or agreed to in writing, shall any Contributor be
* liable to You for damages, including any direct, indirect, special,
* incidental, or consequential damages of any character arising as a
* result of this License or out of the use or inability to use the
* Work (including but not limited to damages for loss of goodwill,
* work stoppage, computer failure or malfunction, or any and all
* other commercial damages or losses), even if such Contributor
* has been advised of the possibility of such damages.

*
* 9. Accepting Warranty or Additional Liability. While redistributing
* the Work or Derivative Works thereof, You may choose to offer,
* and charge a fee for, acceptance of support, warranty, indemnity,
* or other liability obligations and/or rights consistent with this
* License. However, in accepting such obligations, You may act only
* on Your own behalf and on Your sole responsibility, not on behalf
* of any other Contributor, and only if You agree to indemnify,
* defend, and hold each Contributor harmless for any liability

* incurred by, or claims asserted against, such Contributor by reason
* of your accepting any such warranty or additional liability.

*

* END OF TERMS AND CONDITIONS

*

* APPENDIX: How to apply the Apache License to your work.

*

* To apply the Apache License to your work, attach the following
* boilerplate notice, with the fields enclosed by brackets "[]"
* replaced with your own identifying information. (Don't include
* the brackets!) The text should be enclosed in the appropriate
* comment syntax for the file format. We also recommend that a
* file or class name and description of purpose be included on the
* same "printed page" as the copyright notice for easier
* identification within third-party archives.

*

* Copyright [yyyy] [name of copyright owner]

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*/

/* =====

* The Apache Software License, Version 1.1

*

* Copyright (c) 2001 The Apache Software Foundation. All rights
* reserved.

*

* Redistribution and use in source and binary forms, with or without
* modification, are permitted provided that the following conditions
* are met:

*

* 1. Redistributions of source code must retain the above copyright
* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright
* notice, this list of conditions and the following disclaimer in
* the documentation and/or other materials provided with the
* distribution.

*

* 3. The end-user documentation included with the redistribution,
* if any, must include the following acknowledgment:
* "This product includes software developed by the
* Apache Software Foundation (<http://www.apache.org/>)."
* Alternately, this acknowledgment may appear in the software itself,
* if and wherever such third-party acknowledgments normally appear.

* 4. The names "Apache" and "Apache Software Foundation" and
* "Apache BCEL" must not be used to endorse or promote products
* derived from this software without prior written permission. For
* written permission, please contact apache@apache.org.

* 5. Products derived from this software may not be called "Apache",
* "Apache BCEL", nor may "Apache" appear in their name, without
* prior written permission of the Apache Software Foundation.

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====
*

* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.

*/
Copyright (c) 2000-2003 Daisuke Okajima and Kohsuke Kawaguchi.
All rights reserved.

Redistribution and use in source and binary forms, with or without
modification, are permitted provided that the following conditions
are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (<http://relaxngcc.sf.net/>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names of the copyright holders must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact the copyright holders.

5. Products derived from this software may not be called "RELAXNGCC", nor may "RELAXNGCC" appear in their name, without prior written permission of the copyright holders.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
DO NOT TRANSLATE OR LOCALIZE

%%The following software may be included in this product:

XML-NamespacesSupport

Use of any of this software is governed by the terms of the license below:

The "Artistic License"

Preamble

The intent of this document is to state the conditions under which a Package may be copied, such that the Copyright Holder maintains some semblance of artistic control over the development of the package, while giving the users of the package the right to use and distribute the Package in a more-or-less customary fashion, plus the right to make

reasonable modifications.

Definitions:

"Package" refers to the collection of files distributed by the Copyright Holder, and derivatives of that collection of files created through textual modification.

"Standard Version" refers to such a Package if it has not been modified, or has been modified in accordance with the wishes of the Copyright Holder as specified below.

"Copyright Holder" is whoever is named in the copyright or copyrights for the package.

"You" is you, if you're thinking about copying or distributing this Package.

"Reasonable copying fee" is whatever you can justify on the basis of media cost, duplication charges, time of people involved, and so on. (You will not be required to justify it to the Copyright Holder, but only to the computing community at large as a market that must bear the fee.)

"Freely Available" means that no fee is charged for the item itself, though there may be fees involved in handling the item. It also means that recipients of the item may redistribute it under the same conditions they received it.

1. You may make and give away verbatim copies of the source form of the Standard Version of this Package without restriction, provided that you duplicate all of the original copyright notices and associated disclaimers.
2. You may apply bug fixes, portability fixes and other modifications derived from the Public Domain or from the Copyright Holder. A Package modified in such a way shall still be considered the Standard Version.
3. You may otherwise modify your copy of this Package in any way, provided that you insert a prominent notice in each changed file stating how and when you changed that file, and provided that you do at least ONE of the following:
 - a) place your modifications in the Public Domain or otherwise make them Freely Available, such as by posting said modifications to Usenet or an equivalent medium, or placing the modifications on a major archive site such as uunet.uu.net, or by allowing the Copyright Holder to include your modifications in the Standard Version of the Package.

b) use the modified Package only within your corporation or organization.

c) rename any non-standard executables so the names do not conflict with standard executables, which must also be provided, and provide a separate manual page for each non-standard executable that clearly documents how it differs from the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

4. You may distribute the programs of this Package in object code or executable form, provided that you do at least ONE of the following:

a) distribute a Standard Version of the executables and library files, together with instructions (in the manual page or equivalent) on where to get the Standard Version.

b) accompany the distribution with the machine-readable source of the Package with your modifications.

c) give non-standard executables non-standard names, and clearly document the differences in manual pages (or equivalent), together with instructions on where to get the Standard Version.

d) make other distribution arrangements with the Copyright Holder.

5. You may charge a reasonable copying fee for any distribution of this Package. You may charge any fee you choose for support of this Package. You may not charge a fee for this Package itself. However, you may distribute this Package in aggregate with other (possibly commercial) programs as part of a larger (possibly commercial) software distribution provided that you do not advertise this Package as a product of your own. You may embed this Package's interpreter within an executable of yours (by linking); this shall be construed as a mere form of aggregation, provided that the complete Standard Version of the interpreter is so embedded.

6. The scripts and library files supplied as input to or produced as output from the programs of this Package do not automatically fall under the copyright of this Package, but belong to whoever generated them, and may be sold commercially, and may be aggregated with this Package. If such scripts or library files are aggregated with this Package via the so-called "undump" or "unexec" methods of producing a binary executable image, then distribution of such an image shall neither be construed as a distribution of this Package nor shall it fall under the restrictions of Paragraphs 3 and 4, provided that you do not represent such an executable image as a Standard Version of this Package.

7. C subroutines (or comparably compiled subroutines in other languages) supplied by you and linked into this Package in order to emulate subroutines and variables of the language defined by this Package shall not be considered part of this Package, but are the equivalent of input as in Paragraph 6, provided these subroutines do not change the language in any way that would cause it to fail the regression tests for the language.

8. Aggregation of this Package with a commercial distribution is always permitted provided that the use of this Package is embedded; that is, when no overt attempt is made to make this Package's interfaces visible to the end user of the commercial distribution. Such use shall not be construed as a distribution of this Package.

9. The name of the Copyright Holder may not be used to endorse or promote products derived from this software without specific prior written permission.

10. THIS PACKAGE IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

The End

Additional License(s)

Copyright (c) 2001-2005 Robin Berjon. All rights reserved.

%%The following software may be included in this product:
iso-relax.jar

Use of any of this software is governed by the terms of the license below:

The MIT License

Copyright (c)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Copyright 200

Additional License(s)

"copyright" and "license" results in the following hits:

- > * The above copyright notice and this permission notice shall be included
- > * distribute, sublicense, and/or sell copies of the Software, and to

GNU, GPL, LGPL reveals no hit. "?" hits a lot of things but none of them are relevant to the licensing terms.

% The following software may be included in this product:

relaxngDatatype.jar

Use of any of this software is governed by the terms of the license below:

Copyright (c) 2001, Thai Open Source Software Center Ltd, Sun Microsystems.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

Neither the names of the copyright holders nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Additional License(s)

Got the following hits. No hit for GNU, GPL, LGPL.

- > Redistributions of source code must retain the above copyright
- > Neither the names of the copyright holders nor the names of its
- > this license is the BSD license.

%%The following software may be included in this product:

RELAX NG Object Model/Parser

Use of any of this software is governed by the terms of the license below:

The MIT License

Copyright (c)

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so,

subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Additional License(s)

See <https://rngom.dev.java.net/doc/index.html>

%% The following software may be included in this product:

RelaxNGCC

Use of any of this software is governed by the terms of the license below:

Copyright (c) 2000-2003 Daisuke Okajima and Kohsuke Kawaguchi.
All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

"This product includes software developed by Daisuke Okajima and Kohsuke Kawaguchi (<http://relaxngcc.sf.net/>)."

Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

4. The names of the copyright holders must not be used to endorse or promote products derived from this software without prior written

permission. For written permission, please contact the copyright holders.

5. Products derived from this software may not be called "RELAXNGCC", nor may "RELAXNGCC" appear in their name, without prior written permission of the copyright holders.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Additional License(s)

None found

% The following software may be included in this product:

XML Resolver library

Use of any of this software is governed by the terms of the license below:

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the

direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of

this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and

wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor

has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Additional License(s)

All occurrences of copyright, license and (c) refer to the Apache 1.1 license.

No occurrences of GNU, GPL, LGPL.

%%The following software may be included in this product:

Stax API (only)

Use of any of this software is governed by the terms of the license below:

Streaming API for XML (JSR-173) Specification

Reference Implementation

License Agreement

READ THE TERMS OF THIS (THE "AGREEMENT") CAREFULLY BEFORE VIEWING OR USING THE SOFTWARE LICENS ED HEREUNDER. BY VIEWING OR USING THE SOFTWARE, YOU AGREE TO THE TERMS OF THIS AGREEMENT. IF YOU ARE ACCESSING THE SOFTWARE ELECTRONICALLY, INDICATE YOUR ACCEPTANCE OF THESE TERMS BY SELE CTING THE "ACCEPT" BUTTON AT THE END OF THIS AGREEMENT. IF YOU DO NOT AGREE TO ALL THESE TERMS , PROMPTLY RETURN THE UNUSED SOFTWARE TO ORIGINAL CONTRIBUTOR, DEFINED HEREIN.

1.0 DEFINITIONS.

1.1. "BEA" means BEA Systems, Inc., the licensor of the Original Code.

1.2. "Contributor" means BEA and each entity that creates or contributes to the creation of Mo difications.

1.3. "Covered Code" means the Original Code or Modifications or the combination of the Origina l Code and Modifications, in each case including portions thereof and corresponding documentat ion released with the source code.

1.4. "Executable" means Covered Code in any form other than Source Code.

1.5. "FCS" means first commercial shipment of a product.

1.6. "Modifications" means any addition to or deletion from the substance or structure of eith er the Original Code or any previous Modifications. When Covered Code is released as a series of files, a Modification is:

(a) Any addition to or deletion from the contents of a file containing Original Code or previ ous Modifications.

(b) Any new file that contains any part of the Original Code or previous Modifications.

1.7. "Original Code" means Source Code of computer software code Reference Implementation.

1.8. "Patent Claims" means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent for which the grantor has the right to grant a license.

1.9. "Reference Implementation" means the prototype or "proof of concept" implementation of the Specification developed and made available for license by or on behalf of BEA.

1.10. "Source Code" means the preferred form of the Covered Code for making modifications to it, including all modules it contains, plus any associated documentation, interface definition files, scripts used to control compilation and installation of an Executable, or source code differential comparisons against either the Original Code or another well known, available Covered Code of the Contributor's choice.

1.11. "Specification" means the written specification for the Streaming API for XML, Java technology developed pursuant to the Java Community Process.

1.12. "Technology Compatibility Kit" or "TCK" means the documentation, testing tools and test suites associated with the Specification as may be revised by BEA from time to time, that is provided so that an implementer of the Specification may determine if its implementation is compliant with the Specification.

1.13. "You" (or "Your") means an individual or a legal entity exercising rights under, and complying with all of the terms of, this Agreement or a future version of this Agreement issued under Section 6.1. For legal entities, "You" includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, "control" means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the

outstanding shares or beneficial ownership of such entity.

2.0 SOURCE CODE LICENSE.

2.1. Copyright Grant. Subject to the terms of this Agreement, each Contributor hereby grants

You a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare derivative works of, publicly display, publicly perform, distribute and sublicense the Covered Code of such Contributor, if any, and such derivative works, in Source Code and Executable form.

2.2. Patent Grant. Subject to the terms of this Agreement, each Contributor hereby grants You

a non-exclusive, worldwide, royalty-free patent license under the Patent Claims to make, use, sell, offer to sell, import and otherwise transfer the Covered Code prepared and provided by such Contributor, if any, in Source Code and Executable form. This patent license shall apply to the Covered Code if, at the time a Modification is added by the Contributor, such addition of the Modification causes such combination to be covered by the Patent Claims. The patent license shall not apply to any other combinations which include the Modification.

2.3. Conditions to Grants. You understand that although each Contributor grants the licenses

to the Covered Code prepared by it, no assurances are provided by any Contributor that the Covered Code does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to You for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, You hereby assume sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow You to distribute Covered Code, it is Your responsibility to acquire that license before distributing such code.

2.4. Contributors' Representation. Each Contributor represents that to its knowledge it has

sufficient copyright rights in the Covered Code it provides, if any, to grant the copyright license set forth in this Agreement.

3.0 DISTRIBUTION RESTRICTIONS.

3.1. Application of Agreement.

The Modifications which You create or to which You contribute are governed by the terms of this Agreement, including without limitation Section 2.0. The Source Code version of Covered Code may be distributed only under the terms of this Agreement or a future version of this Agreement released under Section 6.1, and You must include a copy of this Agreement with every copy of the Source Code You distribute. You may not offer or impose any terms on any Source Code version that alters or restricts the applicable version of this Agreement or the recipients' rights hereunder. However, You may include an additional document offering the additional rights described in Section 3.3.

3.2. Description of Modifications.

You must cause all Covered Code to which You contribute to contain a file documenting the changes You made to create that Covered Code and the date of any change. You must include a prominent statement that the Modification is derived, directly or indirectly, from Original Code provided by BEA and including the name of BEA in (a) the Source Code, and (b) in any notice in an Executable version or related documentation in which You describe the origin or ownership of the Covered Code.

%%The following software may be included in this product:

XMLWriter

Use of any of this software is governed by the terms of the license below:

XMLWriter IS FREE

I hereby abandon any property rights to XMLWriter 0.1, and release all of the XMLWriter 0.1 source code, compiled code, and documentation contained in this distribution into the Public Domain. XMLWriter comes with NO WARRANTY or guarantee of fitness for any purpose.

David Megginson
david@megginson.com
2000-04-19

Additional License(s)

I grep-ed the source. GNU and GPL has no hits, '?' yields 11 hits but none of them are license related. "copyright" and "license" yield no hits either.

/*

* The Apache Software License, Version 1.1

*

*

* Copyright (c) 1999-2004 The Apache Software Foundation. All rights reserved.

*

* Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

*

* 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

*

* 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:

* "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)."

* Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.

*

* 4. The names "Xerces" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.

*

* 5. Products derived from this software may not be called "Apache",

* nor may "Apache" appear in their name, without prior written
* permission of the Apache Software Foundation.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.

* =====

*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation and was
* originally based on software copyright (c) 1999, International
* Business Machines, Inc., <http://www.ibm.com>. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.

*/

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

1.1. Contributor means each individual or entity that creates or contributes to the creation of Modifications.

1.2. Contributor Version means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable means the Covered Software in any form other than Source Code.

1.5. Initial Developer means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License means this document.

1.8. Licensable means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You (or Your) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, You includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, control means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use, reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either

alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipients rights in the Source Code form from the rights set forth in this License. If You distribute the

Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN AS IS BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms

herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as Participant) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a commercial item, as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of commercial computer software (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and commercial computer software documentation as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software

(except to the extent applicable law, if any, provides otherwise), excluding such jurisdictions conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The GlassFish code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

xml-commons/LICENSE.txt \$Id: LICENSE.txt 226068 2003-07-06 03:27:45Z crossley \$

See README.txt for additional licensing information.

/* =====

* The Apache Software License, Version 1.1

*

* Copyright (c) 2001-2003 The Apache Software Foundation. All rights

* reserved.

*

* Redistribution and use in source and binary forms, with or without

* modification, are permitted provided that the following conditions

* are met:

*

* 1. Redistributions of source code must retain the above copyright

* notice, this list of conditions and the following disclaimer.

*

* 2. Redistributions in binary form must reproduce the above copyright

* notice, this list of conditions and the following disclaimer in

* the documentation and/or other materials provided with the

* distribution.

*

* 3. The end-user documentation included with the redistribution,

* if any, must include the following acknowledgment:

* "This product includes software developed by the

* Apache Software Foundation (<http://www.apache.org/>)."

* Alternately, this acknowledgment may appear in the software itself,
 * if and wherever such third-party acknowledgments normally appear.
 *

* 4. The names "Apache" and "Apache Software Foundation" must
 * not be used to endorse or promote products derived from this
 * software without prior written permission. For written
 * permission, please contact apache@apache.org.
 *

* 5. Products derived from this software may not be called "Apache",
 * nor may "Apache" appear in their name, without prior written
 * permission of the Apache Software Foundation.
 *

* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
 * WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
 * OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
 * DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
 * LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
 * USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
 * ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
 * OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
 * OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
 * SUCH DAMAGE.
 * =====
 *

* This software consists of voluntary contributions made by many
 * individuals on behalf of the Apache Software Foundation. For more
 * information on the Apache Software Foundation, please see
 * <http://www.apache.org/>.
 */

1.334 j2objc-annotations 1.3

1.334.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/RetainedWith.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/Property.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/LoopTranslation.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/ObjectiveCName.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/ReflectionSupport.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/RetainedLocalRef.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/J2ObjCIncompatible.java

No license file was found, but licenses were detected in source scan.

/*

- * Copyright 2012 Google Inc. All Rights Reserved.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/Weak.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/AutoreleasePool.java
- * /opt/cola/permits/1131003150_1612875443.99/0/j2objc-annotations-1-3-sources-3-jar/com/google/j2objc/annotations/WeakOuter.java

1.335 micronaut-redis 5.2.0

1.335.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2020 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/package-info.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/cache/AsyncCacheCommands.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/health/package-info.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/cache/RedisCache.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/session/package-info.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/test/EmbeddedRedisServer.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/cache/expiration/ExpirationAfterWritePolicy.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/NamedRedisServersConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/AbstractRedisConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/cache/AbstractRedisCacheConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/RedisSetting.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
jar/io/micronaut/configuration/lettuce/cache/expiration/ConstantExpirationAfterWritePolicy.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-
```

jar/io/micronaut/configuration/lettuce/cache/package-info.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/cache/RedisCacheConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/session/RedisHttpSessionConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/test/package-info.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/DefaultRedisClientFactory.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/cache/DefaultRedisCacheConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/DefaultRedisConfiguration.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/AbstractRedisClientFactory.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/NamedRedisClientFactory.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/cache/SyncCacheCommands.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/RedisConnectionUtil.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/session/RedisSessionStore.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/DefaultRedisClusterClientFactory.java
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/session/RedisSessionCommands.java
No license file was found, but licenses were detected in source scan.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):
* /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/META-INF/native-image/io.micronaut.redis/redis-lettuce/native-image.properties
No license file was found, but licenses were detected in source scan.

/*
* Copyright 2017-2021 original authors
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <https://www.apache.org/licenses/LICENSE-2.0>
*
*

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/ClientResourcesMutator.java
- * /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/MetricsClientResourceMutator.java
- * /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/ThreadPoolClientResourceMutator.java
- * /opt/cola/permits/1331473606_1653513048.6155212/0/micronaut-redis-lettuce-5-2-0-sources-jar/io/micronaut/configuration/lettuce/health/RedisHealthIndicator.java

1.336 apache-avro 1.11.0

1.336.1 Available under license :

Apache Avro

Copyright 2009-2020 The Apache Software Foundation

This product includes software developed at
The Apache Software Foundation (<http://www.apache.org/>).

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.337 apache-httpcomponents-core-http/2

5.0.2

1.337.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. **Grant of Copyright License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. **Grant of Patent License.** Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. **Redistribution.** You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

Apache HttpComponents Core

Copyright 2005-2020 The Apache Software Foundation

This product includes software developed at

The Apache Software Foundation (<http://www.apache.org/>).

1.338 rxjava 1.3.8

1.338.1 Available under license :

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/util/Pow2.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/Pow2.java

No license file was found, but licenses were detected in source scan.

=====

Copyright 2014 Netflix, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

=====

This package "rx.internal.operators" is for internal implementation details and can change at any time.

It is excluded from the public Javadocs (<http://netflix.github.io/RxJava/javadoc/>) and should not be relied upon by any code.

In short, changes to public signatures of these classes will not be accounted for in the versioning of RxJava.

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/README.md

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2014 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/GenericScheduledExecutorService.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/ActionN.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/OnErrorNotImplementedException.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorToObservableSortedList.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action9.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/BehaviorSubject.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/SafeSubscriber.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorOnBackpressureDrop.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/CompositeException.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/NewThreadScheduler.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeMergeArray.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorMaterialize.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func2.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/GroupedObservable.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTakeUntil.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action4.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeConcat.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTakeLast.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Observer.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/CachedThreadScheduler.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action5.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaErrorHandler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorZip.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/RxRingBuffer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorToObservableList.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/Observers.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeConcatArray.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeMergeIterable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subscriptions/Subscriptions.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subscriptions/BooleanSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSkipLastTimed.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSkip.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action7.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/BlockingOperatorToFuture.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTake.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorUnsubscribeOn.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Subscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action0.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/LinkedList.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/SubjectSubscriptionManager.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTakeLastTimed.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/UnsubscribeFailedException.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/TestScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action1.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/TestObserver.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorAny.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/operators/OperatorScan.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/EventLoopsScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/PublishSubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Scheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaPlugins.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/BlockingOperatorNext.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleOperatorCast.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeDoOnEach.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/BlockingObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/ImmediateScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/ConnectableObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CachedObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/OnErrorFailedException.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func1.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeMergeDelayErrorArray.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorIgnoreElements.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeMergeDelayErrorIterable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/BufferUntilSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/TrampolineScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subscriptions/MultipleAssignmentSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/NotificationLite.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorAsObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeToObservableFuture.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorAll.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorMerge.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/operators/OnSubscribeDefer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorTakeUntilPredicate.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/BackpressureDrainManager.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/subscriptions/SequentialSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/SingleOnSubscribeMap.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/package-
info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorDematerialize.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorDoOnRequest.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OnSubscribeJoin.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OnSubscribeToMap.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/observers/SerializedObserver.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/BlockingOperatorLatest.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/subscriptions/SerialSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action6.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OnSubscribeAutoConnect.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorThrottleFirst.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/observers/TestSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSkipWhile.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func9.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSubscribeOn.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorTimestamp.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/exceptions/MissingBackpressureException.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorReplay.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/subscriptions/CompositeSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorZipIterable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func5.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDelay.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeUsing.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/FuncN.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeSingle.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/SubscriptionList.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorOnBackpressureBuffer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/TestSubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorOnErrorResumeNextViaFunction.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/OnErrorThrowable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/NewThreadScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ScalarSynchronousObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action3.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeRedo.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/SerializedSubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Completable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/TrampolineScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeGroupJoin.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeTimeout.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Subscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/Schedulers.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleDoOnEvent.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFilter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSkipLast.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/UnicastSubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFromArray.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/schedulers/SleepingAction.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action8.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/AsyncSubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/ImmediateScheduler.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTimeInterval.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func7.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDoOnUnsubscribe.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSingle.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeConcatIterable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Functions.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorElementAt.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func0.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/SyncOnSubscribe.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/Subscribers.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeAmb.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeMap.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/TimeInterval.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSequenceEqual.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorMapNotification.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/ReplaySubject.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/IndexedRingBuffer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/BlockingOperatorToIterator.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Notification.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func6.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subscriptions/RefCountSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorCast.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/Exceptions.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeTimeoutSelectorWithFallback.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/OnCompletedFailedException.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Function.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTakeWhile.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeRange.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaObservableExecutionHook.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/BlockingOperatorMostRecent.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func4.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/Subject.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeTimeoutTimedWithFallback.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/SerializedSubscriber.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action2.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorWithLatestFromMany.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorObserveOn.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorSerialize.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func8.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Producer.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaObservableExecutionHookDefault.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Action.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDoOnSubscribe.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFromIterable.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorGroupBy.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/schedulers/Timestamped.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeSwitchIfEmpty.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableOnSubscribeMerge.java
 * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Func3.java
 No license file was found, but licenses were detected in source scan.

This package contains code that relies on sun.misc.Unsafe. Before using it you MUST assert UnsafeAccess.isUnsafeAvailable() == true

Much of the code in this package comes from or is inspired by the JCTools project:
<https://github.com/JCTools/JCTools>

JCTools has now published artifacts (<https://github.com/JCTools/JCTools/issues/17>) so RxJava could add JCTools as a "shadow" dependency (<https://github.com/ReactiveX/RxJava/issues/1735>).

RxJava has a "zero dependency" policy for the core library, so if we do add it as a dependency, it won't be an externally visible dependency that results in a separate jar.

The license for the JCTools code is <https://github.com/JCTools/JCTools/blob/master/LICENSE>

As of June 10 2014 when this code was copied the LICENSE read as:

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or

Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work

or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work

by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/README.md

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2016 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleTimeout.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/operators/OnSubscribeOnAssemblySingle.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleFromObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ObserverSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/AssertableSubscriberObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ActionSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaSingleExecutionHook.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleFromCallable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeLift.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/annotations/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/AsyncCompletableSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaCompletableExecutionHook.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleOnErrorReturn.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeDetach.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleTakeUntilCompletable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFlattenIterable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Cancellable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleTakeUntilObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Emitter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/EmptyObservableHolder.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleDoOnUnsubscribe.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/exceptions/AssemblyStackTraceException.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ActionNotificationObserver.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subscriptions/package-info.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/SchedulePeriodicHelper.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/InternalObservableUtils.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeOnAssemblyCompletable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/BackpressureOverflow.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleObserveOn.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeOnAssembly.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaHooks.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleFromFuture.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleLiftObservableOperator.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/subjects/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleToObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleFromEmitter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleDelay.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/SchedulerWhen.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeThrow.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ActionObserver.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/subscriptions/CancellableSubscription.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleTakeUntilSingle.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/subscriptions/Unsubscribed.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleDoOnSubscribe.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/NeverObservableHolder.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/AssertableSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observers/SafeCompletableSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableFlatMapSingleToCompletable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/singles/package-info.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeCreate.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/plugins/RxJavaSingleExecutionHookDefault.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/GenericScheduledExecutorServiceFactory.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/SpSCUnboundedAtomicArrayQueue.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/SpSCUnboundedAtomicArrayQueue.java
No license file was found, but licenses were detected in source scan.

/**
* Copyright 2018 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
*/

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorGroupByEvicting.java
No license file was found, but licenses were detected in source scan.

/*

- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- *
- * Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
- * Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/ConcurrentCircularArrayQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/ConcurrentCircularArrayQueue.java
No license file was found, but licenses were detected in source scan.

/**

- * Copyright 2014 Netflix, Inc.
- *
- * Licensed under the Apache License, Version 2.0 (the "License");
- * you may not use this file except in compliance with the License.
- * You may obtain a copy of the License at
- *
- * <http://www.apache.org/licenses/LICENSE-2.0>
- *
- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and

* limitations under the License.

*

* Modified from <http://www.javacodegeeks.com/2013/08/simple-and-lightweight-pool-implementation.html>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/OpenHashSet.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/ConcurrentSequencedCircularArrayQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/ConcurrentSequencedCircularArrayQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools->

core/src/main/java/org/jctools/queues/SpmcArrayQueue.java

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/SpmcArrayQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/MessagePassingQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/MessagePassingQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/BaseLinkedQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/BaseLinkedQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/QueueProgressIndicators.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/QueueProgressIndicators.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2016 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeCollect.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/DeferredScalarSubscriberSafe.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/SuppressAnimalSniffer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFromCallable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/SchedulerLifecycle.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/ExceptionsUtils.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeReduceSeed.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeConcatMap.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/DeferredScalarSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleOperatorOnErrorResumeNext.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleOnSubscribeUsing.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeReduce.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleOperatorZip.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeTakeLastOne.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribePublishMulticast.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2015 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not
* use this file except in compliance with the License. You may obtain a copy of
* the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations under
* the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/producers/QueuedProducer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/producers/SingleProducer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/BackpressureUtils.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/producers/ProducerArbiter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/producers/SingleDelayedProducer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/producers/QueuedValueProducer.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/producers/ProducerObserverArbiter.java
No license file was found, but licenses were detected in source scan.

```
/*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software  
* distributed under the License is distributed on an "AS IS" BASIS,  
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.  
* See the License for the specific language governing permissions and  
* limitations under the License.  
*  
* Original License: https://github.com/JCTools/JCTools/blob/master/LICENSE  
* Original location: https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/MpmcArrayQueue.java  
*/
```

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/unsafe/MpmcArrayQueue.java
No license file was found, but licenses were detected in source scan.

```
/**  
* Copyright 2017 Netflix, Inc.  
*  
* Licensed under the Apache License, Version 2.0 (the "License");  
* you may not use this file except in compliance with the License.  
* You may obtain a copy of the License at  
*  
* http://www.apache.org/licenses/LICENSE-2.0  
*  
* Unless required by applicable law or agreed to in writing, software
```

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

*/opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFlatMapCompletable.java

*/opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeFlatMapSingle.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright one 2014 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

*/opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeToMultimap.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2014 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not

* use this file except in compliance with the License. You may obtain a copy of

* the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT

* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the

* License for the specific language governing permissions and limitations under

* the License.

*/

Found in path(s):

- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeTimerPeriodically.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorBufferWithSingleObservable.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeTimerOnce.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/ExecutorScheduler.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorWindowWithObservable.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDistinctUntilChanged.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorMulticast.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeDelaySubscription.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorRetryWithPredicate.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorMapPair.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorPublish.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeSkipTimed.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDelayWithSelector.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/schedulers/ScheduledAction.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorTakeTimed.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorBufferWithTime.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeRefCount.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/SpSCLinkedArrayQueue.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDistinct.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDebounceWithSelector.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OperatorDebounceWithTime.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeDelaySubscriptionWithSelector.java
- * /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/schedulers/NewThreadWorker.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorBufferWithSize.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorWindowWithStartEndObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/SingleOnSubscribeDelaySubscriptionOther.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/PlatformDependent.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/RxThreadFactory.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorBufferWithStartEndObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorDoAfterTerminate.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSwitch.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorWithLatestFrom.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OnSubscribeDelaySubscriptionOther.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorWindowWithSize.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSampleWithObservable.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorOnBackpressureLatest.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorEagerConcatMap.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorWindowWithObservableFactory.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSampleWithTime.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorWindowWithTime.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/functions/Actions.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/operators/OperatorSkipUntil.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/MpscLinkedAtomicQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/MpscLinkedAtomicQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/BaseLinkedAtomicQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/BaseLinkedAtomicQueue.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2014 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in

* compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is

* distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express

or implied. See

* the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/UtilityFunctions.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Observable.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Inspired from <https://code.google.com/p/guava->

libraries/source/browse/guava/src/com/google/common/annotations/Beta.java

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/annotations/Experimental.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: [https://github.com/JCTools/JCTools/blob/master/jctools-](https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java)

core/src/main/java/org/jctools/queues/atomic/AtomicReferenceArrayQueue.java

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/AtomicReferenceArrayQueue.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2015 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/SingleSubscriber.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/singles/BlockingSingle.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/observables/AsyncOnSubscribe.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/SingleDoAfterTerminate.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/BlockingUtils.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2015 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in

* compliance with the License. You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software distributed under the License is

* distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See

* the License for the specific language governing permissions and limitations under the License.

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/OnSubscribeCombineLatest.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/Single.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 Netflix, Inc.
*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/CompletableEmitter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/CompletableFromEmitter.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/CompletableSubscriber.java
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/SingleEmitter.java
No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools->

core/src/main/java/org/jctools/queues/SpSCLinkedQueue.java

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/SpSCLinkedQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/SpSCArrayQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/SpSCArrayQueue.java

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/MpscLinkedQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/MpscLinkedQueue.java

No license file was found, but licenses were detected in source scan.

Copyright 2014 Netflix, Inc.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE>

2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/operators/package.html

No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/SpscAtomicArrayQueue.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/SpscAtomicArrayQueue.java

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-

jar/rx/internal/util/atomic/SpSCExactAtomicArrayQueue.java

No license file was found, but licenses were detected in source scan.

```
/**
```

```
* Copyright 2014 Netflix, Inc.
```

```
*
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
*
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
*
```

```
* Unless required by applicable law or agreed to in writing, software
```

```
* distributed under the License is distributed on an "AS IS" BASIS,
```

```
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
```

```
* See the License for the specific language governing permissions and
```

```
* limitations under the License.
```

```
*/
```

```
/**
```

```
* This plugin class provides 2 ways to customize { @link Scheduler } functionality
```

```
* 1. You may redefine entire schedulers, if you so choose. To do so, override
```

```
* the 3 methods that return Scheduler (io(), computation(), newThread()).
```

```
* 2. You may wrap/decorate an { @link Action0 }, before it is handed off to a Scheduler. The system-
```

```
* supplied Schedulers (Schedulers.ioScheduler, Schedulers.computationScheduler,
```

```
* Scheduler.newThreadScheduler) all use this hook, so it's a convenient way to
```

```
* modify Scheduler functionality without redefining Schedulers wholesale.
```

```
*
```

```
* Also, when redefining Schedulers, you are free to use/not use the onSchedule decoration hook.
```

```
* <p>
```

```
* See { @link RxJavaPlugins } or the RxJava GitHub Wiki for information on configuring plugins:
```

```
* <a
```

```
href="https://github.com/ReactiveX/RxJava/wiki/Plugins">https://github.com/ReactiveX/RxJava/wiki/Plugins</a>.
```

```
*/
```

Found in path(s):

```
* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
```

```
jar/rx/plugins/RxJavaSchedulersHook.java
```

No license file was found, but licenses were detected in source scan.

```
/**
```

```
* Copyright 2014 Netflix, Inc.
```

```
* <p/>
```

```
* Licensed under the Apache License, Version 2.0 (the "License");
```

```
* you may not use this file except in compliance with the License.
```

```
* You may obtain a copy of the License at
```

```
* <p/>
```

```
* http://www.apache.org/licenses/LICENSE-2.0
```

```
* <p/>
```

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/ScalarSynchronousSingle.java

No license file was found, but licenses were detected in source scan.

/**

* Copyright 2014 Netflix, Inc.

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

/*

* This mechanism for getting UNSAFE originally from:

*

* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>

* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/util/UnsafeAccess.java>

*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-
jar/rx/internal/util/unsafe/UnsafeAccess.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright (C) 2010 The Guava Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.

*
* Originally from <https://code.google.com/p/guava-libraries/source/browse/guava/src/com/google/common/annotations/Beta.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/annotations/Beta.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/SpSCUnboundedArrayQueue.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/unsafe/SpSCUnboundedArrayQueue.java
No license file was found, but licenses were detected in source scan.

/*
* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/atomic/SpScLinkedAtomicQueue.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/SpScLinkedAtomicQueue.java
No license file was found, but licenses were detected in source scan.

/*

* Licensed under the Apache License, Version 2.0 (the "License");
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at
*
* <http://www.apache.org/licenses/LICENSE-2.0>
*
* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
* Original License: <https://github.com/JCTools/JCTools/blob/master/LICENSE>
* Original location: <https://github.com/JCTools/JCTools/blob/master/jctools-core/src/main/java/org/jctools/queues/LinkedQueueNode.java>
*/

Found in path(s):

* /opt/ws_local/PERMITS_SQL/1014221113_1591382734.95/0/rxjava-1-3-8-sources-jar/rx/internal/util/atomic/LinkedQueueNode.java

1.339 jakarta-mail 1.5.0-b01

1.339.1 Available under license :

COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0

1. Definitions.

- 1.1. Contributor. means each individual or entity that creates or contributes to the creation of Modifications.
- 1.2. Contributor Version. means the combination of the Original Software, prior Modifications used by a Contributor (if any), and the Modifications made by that particular Contributor.

1.3. Covered Software. means (a) the Original Software, or (b) Modifications, or (c) the combination of files containing Original Software with files containing Modifications, in each case including portions thereof.

1.4. Executable. means the Covered Software in any form other than Source Code.

1.5. Initial Developer. means the individual or entity that first makes Original Software available under this License.

1.6. Larger Work. means a work which combines Covered Software or portions thereof with code not governed by the terms of this License.

1.7. License. means this document.

1.8. Licensable. means having the right to grant, to the maximum extent possible, whether at the time of the initial grant or subsequently acquired, any and all of the rights conveyed herein.

1.9. Modifications. means the Source Code and Executable form of any of the following:

A. Any file that results from an addition to, deletion from or modification of the contents of a file containing Original Software or previous Modifications;

B. Any new file that contains any part of the Original Software or previous Modification; or

C. Any new file that is contributed or otherwise made available under the terms of this License.

1.10. Original Software. means the Source Code and Executable form of computer software code that is originally released under this License.

1.11. Patent Claims. means any patent claim(s), now owned or hereafter acquired, including without limitation, method, process, and apparatus claims, in any patent Licensable by grantor.

1.12. Source Code. means (a) the common form of computer software code in which modifications are made and (b) associated documentation included in or with such code.

1.13. You. (or .Your.) means an individual or a legal entity exercising rights under, and complying with all of the terms of, this License. For legal entities, .You. includes any entity which controls, is controlled by, or is under common control with You. For purposes of this definition, .control. means (a) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (b) ownership of more than fifty percent (50%) of the outstanding shares or beneficial ownership of such entity.

2. License Grants.

2.1. The Initial Developer Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, the Initial Developer hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Initial Developer, to use,

reproduce, modify, display, perform, sublicense and distribute the Original Software (or portions thereof), with or without Modifications, and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using or selling of Original Software, to make, have made, use, practice, sell, and offer for sale, and/or otherwise dispose of the Original Software (or portions thereof).

(c) The licenses granted in Sections 2.1(a) and (b) are effective on the date Initial Developer first distributes or otherwise makes the Original Software available to a third party under the terms of this License.

(d) Notwithstanding Section 2.1(b) above, no patent license is granted: (1) for code that You delete from the Original Software, or (2) for infringements caused by: (i) the modification of the Original Software, or (ii) the combination of the Original Software with other software or devices.

2.2. Contributor Grant.

Conditioned upon Your compliance with Section 3.1 below and subject to third party intellectual property claims, each Contributor hereby grants You a world-wide, royalty-free, non-exclusive license:

(a) under intellectual property rights (other than patent or trademark) Licensable by Contributor to use, reproduce, modify, display, perform, sublicense and distribute the Modifications created by such Contributor (or portions thereof), either on an unmodified basis, with other Modifications, as Covered Software and/or as part of a Larger Work; and

(b) under Patent Claims infringed by the making, using, or selling of Modifications made by that Contributor either alone and/or in combination with its Contributor Version (or portions of such combination), to make, use, sell, offer for sale, have made, and/or otherwise dispose of: (1) Modifications made by that Contributor (or portions thereof); and (2) the combination of Modifications made by that Contributor with its Contributor Version (or portions of such combination).

(c) The licenses granted in Sections 2.2(a) and 2.2(b) are effective on the date Contributor first distributes or otherwise makes the Modifications available to a third party.

(d) Notwithstanding Section 2.2(b) above, no patent license is granted: (1) for any code that Contributor has deleted from the Contributor Version; (2) for infringements caused by: (i) third party modifications of Contributor Version, or (ii) the combination of Modifications made by that Contributor with other software (except as part of the Contributor Version) or other devices; or (3) under Patent Claims infringed by Covered Software in the absence of Modifications made by that Contributor.

3. Distribution Obligations.

3.1. Availability of Source Code.

Any Covered Software that You distribute or otherwise make available in Executable form must also be made available in Source Code form and that Source Code form must be distributed only under the terms of this License. You must include a copy of this License with every copy of the Source Code form of the Covered Software You distribute or otherwise make available. You must inform recipients of any such Covered Software in Executable form as to how they can obtain such Covered Software in Source Code form in a reasonable manner on or through a medium customarily used for software exchange.

3.2. Modifications.

The Modifications that You create or to which You contribute are governed by the terms of this License. You represent that You believe Your Modifications are Your original creation(s) and/or You have sufficient rights to grant the rights conveyed by this License.

3.3. Required Notices.

You must include a notice in each of Your Modifications that identifies You as the Contributor of the Modification. You may not remove or alter any copyright, patent or trademark notices contained within the Covered Software, or any notices of licensing or any descriptive text giving attribution to any Contributor or the Initial Developer.

3.4. Application of Additional Terms.

You may not offer or impose any terms on any Covered Software in Source Code form that alters or restricts the applicable version of this License or the recipients' rights hereunder. You may choose to offer, and to charge a fee for, warranty, support, indemnity or liability obligations to one or more recipients of Covered Software. However, you may do so only on Your own behalf, and not on behalf of the Initial Developer or any Contributor. You must make it absolutely clear that any such warranty, support, indemnity or liability obligation is offered by You alone, and You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of warranty, support, indemnity or liability terms You offer.

3.5. Distribution of Executable Versions.

You may distribute the Executable form of the Covered Software under the terms of this License or under the terms of a license of Your choice, which may contain terms different from this License, provided that You are in compliance with the terms of this License and that the license for the Executable form does not attempt to limit or alter the recipient's rights in the Source Code form from the rights set forth in this License. If You distribute the Covered Software in Executable form under a different license, You must make it absolutely clear that any terms which differ from this License are offered by You alone, not by the Initial Developer or Contributor. You hereby agree to indemnify the Initial Developer and every Contributor for any liability incurred by the Initial Developer or such Contributor as a result of any such terms You offer.

3.6. Larger Works.

You may create a Larger Work by combining Covered Software with other code not governed by the terms of this License and distribute the Larger Work as a single product. In such a case, You must make sure the requirements of this License are fulfilled for the Covered Software.

4. Versions of the License.

4.1. New Versions.

Sun Microsystems, Inc. is the initial license steward and may publish revised and/or new versions of this License from time to time. Each version will be given a distinguishing version number. Except as provided in Section 4.3, no one other than the license steward has the right to modify this License.

4.2. Effect of New Versions.

You may always continue to use, distribute or otherwise make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. If the Initial Developer includes a notice in the Original Software prohibiting it from being distributed or otherwise made available under any subsequent version of the License, You must distribute and make the Covered Software available under the terms of the version of the License under which You originally received the Covered Software. Otherwise, You may

also choose to use, distribute or otherwise make the Covered Software available under the terms of any subsequent version of the License published by the license steward.

4.3. Modified Versions.

When You are an Initial Developer and You want to create a new license for Your Original Software, You may create and use a modified version of this License if You: (a) rename the license and remove any references to the name of the license steward (except to note that the license differs from this License); and (b) otherwise make it clear that the license contains terms which differ from this License.

5. DISCLAIMER OF WARRANTY.

COVERED SOFTWARE IS PROVIDED UNDER THIS LICENSE ON AN .AS IS. BASIS, WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, WARRANTIES THAT THE COVERED SOFTWARE IS FREE OF DEFECTS, MERCHANTABILITY, FIT FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE COVERED SOFTWARE IS WITH YOU. SHOULD ANY COVERED SOFTWARE PROVE DEFECTIVE IN ANY RESPECT, YOU (NOT THE INITIAL DEVELOPER OR ANY OTHER CONTRIBUTOR) ASSUME THE COST OF ANY NECESSARY SERVICING, REPAIR OR CORRECTION. THIS DISCLAIMER OF WARRANTY CONSTITUTES AN ESSENTIAL PART OF THIS LICENSE. NO USE OF ANY COVERED SOFTWARE IS AUTHORIZED HEREUNDER EXCEPT UNDER THIS DISCLAIMER.

6. TERMINATION.

6.1. This License and the rights granted hereunder will terminate automatically if You fail to comply with terms herein and fail to cure such breach within 30 days of becoming aware of the breach. Provisions which, by their nature, must remain in effect beyond the termination of this License shall survive.

6.2. If You assert a patent infringement claim (excluding declaratory judgment actions) against Initial Developer or a Contributor (the Initial Developer or Contributor against whom You assert such claim is referred to as .Participant.) alleging that the Participant Software (meaning the Contributor Version where the Participant is a Contributor or the Original Software where the Participant is the Initial Developer) directly or indirectly infringes any patent, then any and all rights granted directly or indirectly to You by such Participant, the Initial Developer (if the Initial Developer is not the Participant) and all Contributors under Sections 2.1 and/or 2.2 of this License shall, upon 60 days notice from Participant terminate prospectively and automatically at the expiration of such 60 day notice period, unless if within such 60 day period You withdraw Your claim with respect to the Participant Software against such Participant either unilaterally or pursuant to a written agreement with Participant.

6.3. In the event of termination under Sections 6.1 or 6.2 above, all end user licenses that have been validly granted by You or any distributor hereunder prior to termination (excluding licenses granted to You by any distributor) shall survive termination.

7. LIMITATION OF LIABILITY.

UNDER NO CIRCUMSTANCES AND UNDER NO LEGAL THEORY, WHETHER TORT (INCLUDING NEGLIGENCE), CONTRACT, OR OTHERWISE, SHALL YOU, THE INITIAL DEVELOPER, ANY OTHER CONTRIBUTOR, OR ANY DISTRIBUTOR OF COVERED SOFTWARE, OR ANY SUPPLIER OF ANY OF SUCH PARTIES, BE LIABLE TO ANY PERSON FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY CHARACTER INCLUDING, WITHOUT LIMITATION, DAMAGES

FOR LOST PROFITS, LOSS OF GOODWILL, WORK STOPPAGE, COMPUTER FAILURE OR MALFUNCTION, OR ANY AND ALL OTHER COMMERCIAL DAMAGES OR LOSSES, EVEN IF SUCH PARTY SHALL HAVE BEEN INFORMED OF THE POSSIBILITY OF SUCH DAMAGES. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO LIABILITY FOR DEATH OR PERSONAL INJURY RESULTING FROM SUCH PARTY'S NEGLIGENCE TO THE EXTENT APPLICABLE LAW PROHIBITS SUCH LIMITATION. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THIS EXCLUSION AND LIMITATION MAY NOT APPLY TO YOU.

8. U.S. GOVERNMENT END USERS.

The Covered Software is a .commercial item., as that term is defined in 48 C.F.R. 2.101 (Oct. 1995), consisting of .commercial computer software. (as that term is defined at 48 C.F.R. 252.227-7014(a)(1)) and .commercial computer software documentation. as such terms are used in 48 C.F.R. 12.212 (Sept. 1995). Consistent with 48 C.F.R. 12.212 and 48 C.F.R. 227.7202-1 through 227.7202-4 (June 1995), all U.S. Government End Users acquire Covered Software with only those rights set forth herein. This U.S. Government Rights clause is in lieu of, and supersedes, any other FAR, DFAR, or other clause or provision that addresses Government rights in computer software under this License.

9. MISCELLANEOUS.

This License represents the complete agreement concerning subject matter hereof. If any provision of this License is held to be unenforceable, such provision shall be reformed only to the extent necessary to make it enforceable. This License shall be governed by the law of the jurisdiction specified in a notice contained within the Original Software (except to the extent applicable law, if any, provides otherwise), excluding such jurisdiction's conflict-of-law provisions. Any litigation relating to this License shall be subject to the jurisdiction of the courts located in the jurisdiction and venue specified in a notice contained within the Original Software, with the losing party responsible for costs, including, without limitation, court costs and reasonable attorneys' fees and expenses. The application of the United Nations Convention on Contracts for the International Sale of Goods is expressly excluded. Any law or regulation which provides that the language of a contract shall be construed against the drafter shall not apply to this License. You agree that You alone are responsible for compliance with the United States export administration regulations (and the export control laws and regulation of any other countries) when You use, distribute or otherwise make available any Covered Software.

10. RESPONSIBILITY FOR CLAIMS.

As between Initial Developer and the Contributors, each party is responsible for claims and damages arising, directly or indirectly, out of its utilization of rights under this License and You agree to work with Initial Developer and Contributors to distribute such responsibility on an equitable basis. Nothing herein is intended or shall be deemed to constitute any admission of liability.

NOTICE PURSUANT TO SECTION 9 OF THE COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL)

The code released under the CDDL shall be governed by the laws of the State of California (excluding conflict-of-law provisions). Any litigation relating to this License shall be subject to the jurisdiction of the Federal Courts of the Northern District of California and the state courts of the State of California, with venue lying in Santa Clara County, California.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited

to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE

PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C)

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

Gnomovision version 69, Copyright (C) year name of author

Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type `show w'. This is free software, and you are welcome to redistribute it under certain conditions; type `show c' for details.

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License.

Of course, the commands you use may be called something other than `show w` and `show c`; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

"CLASSPATH" EXCEPTION TO THE GPL VERSION 2

Certain source files distributed by Sun Microsystems, Inc. are subject to the following clarification and special exception to the GPL Version 2, but only where Sun has expressly included in the particular source file's header the words

"Sun designates this particular file as subject to the "Classpath" exception as provided by Sun in the License file that accompanied this code."

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License Version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module.? An independent module is a module which is not derived from or based on this library.? If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so.? If you do not wish to do so, delete this exception statement from your version.

1.340 fabric8-:::-kubernetes-model-:::- extensions 4.10.3

1.340.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted"

means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and

attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the

appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software

distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.341 kafka-schema-serializer 6.2.0

1.341.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="http://maven.apache.org/POM/4.0.0"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <parent>
    <groupId>io.confluent</groupId>
    <artifactId>kafka-schema-registry-parent</artifactId>
    <version>6.2.0</version>
  </parent>

  <licenses>
    <license>
      <name>Apache License 2.0</name>
      <url>http://www.apache.org/licenses/LICENSE-2.0.html</url>
      <distribution>repo</distribution>
    </license>
  </licenses>

  <artifactId>kafka-schema-serializer</artifactId>
  <packaging>jar</packaging>
  <name>kafka-schema-serializer</name>

  <dependencies>
    <dependency>
      <groupId>org.apache.kafka</groupId>
      <artifactId>kafka_${kafka.scala.version}</artifactId>
      <scope>provided</scope>
    </dependency>
    <dependency>
      <groupId>io.confluent</groupId>
      <artifactId>kafka-schema-registry-client</artifactId>
      <version>${io.confluent.schema-registry.version}</version>
    </dependency>
    <dependency>
      <groupId>junit</groupId>
```



```
<artifactId>junit</artifactId>
  <scope>test</scope>
</dependency>
</dependencies>
</project>
```

Found in path(s):

```
* /opt/cola/permits/1340188452_1654734909.4941463/0/kafka-schema-serializer-6-2-0-jar/META-INF/maven/io.confluent/kafka-schema-serializer/pom.xml
```

1.342 zjsonpatch 0.3.0

1.342.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
```

```
<groupId>com.flipkart.zjsonpatch</groupId>
<artifactId>zjsonpatch</artifactId>
<version>0.3.0</version>
<packaging>jar</packaging>
```

```
<name>zjsonpatch</name>
<description>Java Library to find / apply JSON Patches according to RFC 6902</description>
<url>https://github.com/flipkart-incubator/zjsonpatch</url>
```

```
<scm>
  <connection>scm:git:https://github.com/flipkart-incubator/zjsonpatch.git</connection>
  <developerConnection>scm:git:https://github.com/flipkart-incubator/zjsonpatch.git</developerConnection>
  <tag>HEAD</tag>
  <url>https://github.com/flipkart-incubator/zjsonpatch</url>
</scm>
```

```
<developers>
  <developer>
    <id>vishwakarma</id>
    <name>Gopi Vishwakarma</name>
    <email>vishwakarma.iiita@gmail.com</email>
  </developer>
</developers>
```

```
<licenses>
  <license>
    <name>The Apache Software License, Version 2.0</name>
    <url>http://www.apache.org/licenses/LICENSE-2.0.txt</url>
```

```

    <distribution>repo</distribution>
  </license>
</licenses>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <jackson.version>2.8.5</jackson.version>
</properties>

<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>2.3.1</version>
      <configuration>
        <source>1.6</source>
        <target>1.6</target>
        <showDeprecation>>true</showDeprecation>
        <showWarnings>>true</showWarnings>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-source-plugin</artifactId>
      <version>2.2.1</version>
      <executions>
        <execution>
          <id>attach-sources</id>
          <goals>
            <goal>jar-no-fork</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-javadoc-plugin</artifactId>
      <version>2.9.1</version>
      <executions>
        <execution>
          <id>attach-javadocs</id>
          <goals>
            <goal>jar</goal>
          </goals>
        </execution>
      </executions>
    </plugin>
  </plugins>
</build>

```

```

<plugin>
  <groupId>org.sonatype.plugins</groupId>
  <artifactId>nexus-staging-maven-plugin</artifactId>
  <version>1.6.3</version>
  <extensions>true</extensions>
  <configuration>
    <serverId>osrh</serverId>
    <nexusUrl>https://oss.sonatype.org/</nexusUrl>
    <autoReleaseAfterClose>true</autoReleaseAfterClose>
  </configuration>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-gpg-plugin</artifactId>
  <version>1.4</version>
  <executions>
    <execution>
      <id>sign-artifacts</id>
      <phase>verify</phase>
      <goals>
        <goal>sign</goal>
      </goals>
    </execution>
  </executions>
</plugin>
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-release-plugin</artifactId>
  <version>2.5</version>
  <configuration>
    <autoVersionSubmodules>true</autoVersionSubmodules>
    <useReleaseProfile>false</useReleaseProfile>
    <releaseProfiles>release</releaseProfiles>
    <goals>deploy</goals>
  </configuration>
</plugin>
</plugins>
</build>
<dependencies>
  <dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-databind</artifactId>
    <version>${jackson.version}</version>
  </dependency>
  <dependency>
    <groupId>com.fasterxml.jackson.core</groupId>
    <artifactId>jackson-core</artifactId>
    <version>${jackson.version}</version>

```

```

</dependency>
<dependency>
  <groupId>com.google.guava</groupId>
  <artifactId>guava</artifactId>
  <version>18.0</version>
</dependency>
<dependency>
  <scope>test</scope>
  <groupId>commons-io</groupId>
  <artifactId>commons-io</artifactId>
  <version>2.4</version>
</dependency>
<dependency>
  <scope>test</scope>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.12</version>
</dependency>
<dependency>
  <groupId>org.apache.commons</groupId>
  <artifactId>commons-collections4</artifactId>
  <version>4.1 </version>
</dependency>
</dependencies>
<distributionManagement>
  <snapshotRepository>
    <id>ossrh</id>
    <url>https://oss.sonatype.org/content/repositories/snapshots</url>
  </snapshotRepository>
  <repository>
    <id>ossrh</id>
    <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
  </repository>
</distributionManagement>
</project>

```

Found in path(s):

* /opt/cola/permits/1288519755_1647351836.02/0/zjsonpatch-0-3-0-jar/META-INF/maven/com.flipkart.zjsonpatch/zjsonpatch/pom.xml

1.343 guava-internalfuturefailureaccess-and-internalfutures 1.0.1

1.343.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright (C) 2018 The Guava Authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except
 * in compliance with the License. You may obtain a copy of the License at
 *
 * http://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software distributed under the License
 * is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express
 * or implied. See the License for the specific language governing permissions and limitations under
 * the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1130987386_1612872111.26/0/failureaccess-1-0-1-sources-
jar/com/google/common/util/concurrent/internal/InternalFutureFailureAccess.java
* /opt/cola/permits/1130987386_1612872111.26/0/failureaccess-1-0-1-sources-
jar/com/google/common/util/concurrent/internal/InternalFutures.java
```

1.344 jetty-toolchain 1.0.4

1.344.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
// All rights reserved. This program and the accompanying materials
// are made available under the terms of the Eclipse Public License v1.0
// and Apache License v2.0 which accompanies this distribution.
// The Eclipse Public License is available at
// The Apache License v2.0 is available at
// You may elect to redistribute this code under either of these licenses.
```

Found in path(s):

```
* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-
jar/org/eclipse/jetty/setuid/Group.java
* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-
jar/org/eclipse/jetty/setuid/SetUIDServer.java
* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-
jar/org/eclipse/jetty/setuid/SetUID.java
* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-
jar/org/eclipse/jetty/setuid/Passwd.java
* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-
jar/org/eclipse/jetty/setuid/RLimit.java
```

* /opt/cola/permits/1293852964_1649804864.87/0/jetty-setuid-java-1-0-4-sources-jar/org/eclipse/jetty/setuid/SetUIDListener.java

1.345 netty/transport/classes/epoll

4.1.74.Final

1.345.1 Available under license :

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2015 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollServerDomainSocketChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollRecvByteAllocatorHandle.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/AbstractEpollStreamChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollDomainSocketChannelConfig.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollServerChannelConfig.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollEventArray.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollMode.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/TcpMd5Util.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollRecvByteAllocatorStreamingHandle.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/AbstractEpollServerChannel.java
```

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollChannelConfig.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollDomainSocketChannel.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2016 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/LinuxSocket.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/NativeStaticallyReferencedJniMethods.java
No license file was found, but licenses were detected in source scan.

/*

* Copyright 2013 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/Native.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2012 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-
jar/io/netty/channel/epoll/EpollDatagramChannelConfig.java
```

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
```

```
/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
```

Found in path(s):

```
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-
jar/io/netty/channel/epoll/EpollServerSocketChannelConfig.java
```


No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2014 The Netty Project
 *
 * The Netty Project licenses this file to you under the Apache License,
 * version 2.0 (the "License"); you may not use this file except in compliance
 * with the License. You may obtain a copy of the License at:
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
 * WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
 * License for the specific language governing permissions and limitations
 * under the License.
 */
/**
 * Set the {@code TCP_MD5SIG} option on the socket. See {@code linux/tcp.h} for more details.
 * Keys can only be set on, not read to prevent a potential leak, as they are confidential.
 * Allowing them being read would mean anyone with access to the channel could get them.
 */
/**
 * Set the {@code TCP_QUICKACK} option on the socket.
 * See <a href="https://linux.die.net/man/7/tcp">TCP_QUICKACK</a>
 * for more details.
 */
```

Found in path(s):

```
*/opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-
jar/io/netty/channel/epoll/EpollSocketChannelConfig.java
```

No license file was found, but licenses were detected in source scan.

```
<!--
~ Copyright 2021 The Netty Project
~
~ The Netty Project licenses this file to you under the Apache License,
~ version 2.0 (the "License"); you may not use this file except in compliance
~ with the License. You may obtain a copy of the License at:
~
~ https://www.apache.org/licenses/LICENSE-2.0
~
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
~ WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
~ License for the specific language governing permissions and limitations
~ under the License.
-->
```

Found in path(s):

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/META-INF/maven/io.netty/netty-transport-classes-epoll/pom.xml

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2021 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollDomainDatagramChannelConfig.java

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollDomainDatagramChannel.java

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/SegmentedDatagramPacket.java

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2014 The Netty Project

*

* The Netty Project licenses this file to you under the Apache License,
* version 2.0 (the "License"); you may not use this file except in compliance
* with the License. You may obtain a copy of the License at:

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an "AS IS" BASIS, WITHOUT
* WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the
* License for the specific language governing permissions and limitations
* under the License.

*/

Found in path(s):

* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollDatagramChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/AbstractEpollChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollChannelOption.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollServerSocketChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/Epoll.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollSocketChannel.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollEventLoop.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/package-info.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollTcpInfo.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/EpollEventLoopGroup.java
* /opt/cola/permits/1340188589_1654733200.8845074/0/netty-transport-classes-epoll-4-1-74-final-sources-2-jar/io/netty/channel/epoll/NativeDatagramPacketArray.java

1.346 ktor-http-jvm 1.2.6

1.346.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the

outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable

copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and

do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.347 utils 5.5.1

1.347.1 Available under license :

MIT License

Copyright (c) 2016 MobX

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal

in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.348 javax-inject1-as-osgi-bundle 2.6.1

1.348.1 Available under license :

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition

of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in

writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the

minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code.

And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.
- b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.
- c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it,

under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

- a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,
- c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are

prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those

countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

Yoyodyne, Inc., hereby disclaims all copyright interest in the program `Gnomovision' (which makes passes at compilers) written by James Hacker.

signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Notices for Eclipse GlassFish

This content is produced and maintained by the Eclipse GlassFish project.

* Project home: <https://projects.eclipse.org/projects/ee4j.glassfish>

Trademarks

Eclipse GlassFish, and GlassFish are trademarks of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/glassfish-ha-api>
- * <https://github.com/eclipse-ee4j/glassfish-logging-annotation-processor>
- * <https://github.com/eclipse-ee4j/glassfish-shoal>
- * <https://github.com/eclipse-ee4j/glassfish-cdi-porting-tck>
- * <https://github.com/eclipse-ee4j/glassfish-jsftemplating>
- * <https://github.com/eclipse-ee4j/glassfish-hk2-extra>
- * <https://github.com/eclipse-ee4j/glassfish-hk2>
- * <https://github.com/eclipse-ee4j/glassfish-fighterfish>

Third-party Content

This project leverages the following third party content.

None

Cryptography

Content may contain encryption software. The country in which you are currently may have restrictions on the import, possession, and use, and/or re-export to another country, of encryption software. BEFORE using any encryption software, please check the country's laws, regulations and policies concerning the import, possession, or use, and re-export of encryption software, to see if this is permitted.

1.349 cloudevents---core 2.1.1

1.349.1 Available under license :

No license file was found, but licenses were detected in source scan.

<!--

~ Copyright 2018-Present The CloudEvents Authors
~ <p>
~ Licensed under the Apache License, Version 2.0 (the "License");
~ you may not use this file except in compliance with the License.
~ You may obtain a copy of the License at
~ <p>
~ <http://www.apache.org/licenses/LICENSE-2.0>
~ <p>
~ Unless required by applicable law or agreed to in writing, software
~ distributed under the License is distributed on an "AS IS" BASIS,
~ WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
~ See the License for the specific language governing permissions and
~ limitations under the License.
~
-->

Found in path(s):

* /opt/cola/permits/1337070066_1654221700.3755271/0/cloudevents-core-2-1-1-jar/META-INF/maven/io.cloudevents/cloudevents-core/pom.xml

1.350 metrics-core 4.1.17

1.350.1 Available under license :

Apache-2.0

1.351 io-grpc-grpc-api 1.39.0

1.351.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition,

"control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or,

within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all

other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2016-2020 Istio Authors

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

Envoy

Copyright 2016-2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner

or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.
6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions

of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner].

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

protoc-gen-validate

Copyright 2019 Envoy Project Authors

Licensed under Apache License 2.0. See LICENSE for terms.

zero-allocation-hashing

Copyright 2015 Higher Frequency Trading <http://www.higherfrequencytrading.com>

Licensed under Apache License 2.0. See LICENSE for terms.

Apache License

Version 2.0, January 2004

<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a

cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise,

any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.
7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

/*

* Copyright 2015 The gRPC Authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <http://www.apache.org/licenses/LICENSE-2.0>

*

* Unless required by applicable law or agreed to in writing, software

* distributed under the License is distributed on an "AS IS" BASIS,

* WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

* See the License for the specific language governing permissions and

* limitations under the License.

*/

Copyright 2014 The gRPC Authors

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS,

WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

This product contains a modified portion of 'OkHttp', an open source
HTTP & SPDY client for Android and Java applications, which can be obtained
at:

- * LICENSE:
 - * okhttp/third_party/okhttp/LICENSE (Apache License 2.0)
- * HOMEPAGE:
 - * <https://github.com/square/okhttp>
- * LOCATION_IN_GRPC:
 - * okhttp/third_party/okhttp

This product contains a modified portion of 'Envoy', an open source
cloud-native high-performance edge/middle/service proxy, which can be
obtained at:

- * LICENSE:
 - * xds/third_party/envoy/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/envoy/NOTICE
- * HOMEPAGE:
 - * <https://www.envoyproxy.io>
- * LOCATION_IN_GRPC:
 - * xds/third_party/envoy

This product contains a modified portion of 'protoc-gen-validate (PGV)',
an open source protoc plugin to generate polyglot message validators,
which can be obtained at:

- * LICENSE:
 - * xds/third_party/protoc-gen-validate/LICENSE (Apache License 2.0)
- * NOTICE:
 - * xds/third_party/protoc-gen-validate/NOTICE
- * HOMEPAGE:
 - * <https://github.com/envoyproxy/protoc-gen-validate>
- * LOCATION_IN_GRPC:
 - * xds/third_party/protoc-gen-validate

This product contains a modified portion of 'udpa',
an open source universal data plane API, which can be obtained at:

- * LICENSE:
 - * xds/third_party/udpa/LICENSE (Apache License 2.0)

- * HOMEPAGE:
 - * <https://github.com/cncf/udpa>
- * LOCATION_IN_GRPC:
 - * xds/third_party/udpa

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes

of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You

meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor,

except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.
8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.
9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

1.352 apiguardian-apiguardian-api 1.1.0

1.352.1 Available under license :

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction,
and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by
the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all
other entities that control, are controlled by, or are under common
control with that entity. For the purposes of this definition,
"control" means (i) the power, direct or indirect, to cause the
direction or management of such entity, whether by contract or
otherwise, or (ii) ownership of fifty percent (50%) or more of the
outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity
exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications,
including but not limited to software source code, documentation
source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work,

where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
- (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
- (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or

for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason

of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "{}" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.353 dropwizard-configuration-support 2.0.18

1.353.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems,

and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. **Submission of Contributions.** Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. **Trademarks.** This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. **Disclaimer of Warranty.** Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and

limitations under the License.

1.354 junit-5-bill-of-materials 5.7.2

1.354.1 Available under license :

```
import java.io.File
import java.net.URI
```

```
data class License(val name: String, val url: URI, val headerFile: File)
Apache License
=====
```

```
_Version 2.0, January 2004_
_&lt;<https://www.apache.org/licenses/>&gt;_
```

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * **(a)** You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * **(b)** You must cause any modified files to carry prominent notices stating that You changed the files; and
- * **(c)** You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * **(d)** If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

Eclipse Public License - v 2.0

=====

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE (AGREEMENT). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. Definitions

Contribution means:

- * **a)** in the case of the initial Contributor, the initial content Distributed under this Agreement, and
- * **b)** in the case of each subsequent Contributor:
 - * **i)** changes to the Program, and
 - * **j)** additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution originates from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

Contributor means any person or entity that Distributes the Program.

Licensed Patents mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

Program means the Contributions Distributed in accordance with this Agreement.

Recipient means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

Derivative Works shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

Modified Works shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

Distribute means the acts of ****a)**** distributing or ****b)**** making available in any manner that enables the transfer of a copy.

Source Code means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

Secondary License means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. Grant of Rights

****a)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

****b)**** Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

****c)**** Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein,

no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

****d)**** Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

****e)**** Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. Requirements

****3.1)**** If a Contributor Distributes the Program in any form, then:

****a)**** the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

****b)**** the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

****i)**** effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

****ii)**** effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

****iii)**** does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

****iv)**** requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

****3.2)**** When the Program is Distributed as Source Code:

****a)**** it must be made available under this Agreement, or if the Program ****i)**** is combined with other material in a separate file or files made available under a Secondary License, and ****ii)**** the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and

****b)**** a copy of this Agreement must be included with each copy of the Program.

****3.3)**** Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability (notices) contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. Commercial Distribution

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor (Commercial Contributor) hereby agrees to defend and indemnify every other Contributor (Indemnified Contributor) against any losses, damages and costs (collectively Losses) arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: ****a)**** promptly notify the Commercial Contributor in writing of such claim, and ****b)**** allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. No Warranty

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. Disclaimer of Liability

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. General

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

> This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here }.

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

Apache License

=====

Version 2.0, January 2004

<<<https://www.apache.org/licenses/>>>>

Terms and Conditions for use, reproduction, and distribution

1. Definitions

License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means **(i)** the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or **(ii)** ownership of fifty percent (50%) or more of the outstanding shares, or **(iii)** beneficial ownership of such entity.

You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication

that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License

Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution

You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

- * *(a)* You must give any other recipients of the Work or Derivative Works a copy of this License; and
- * *(b)* You must cause any modified files to carry prominent notices stating that You changed the files; and
- * *(c)* You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
- * *(d)* If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the

following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions

Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks

This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty

Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability

In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be

liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability

While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets `[]` replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same printed page as the copyright notice for easier identification within third-party archives.

Copyright [yyyy] [name of copyright owner]

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<https://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.

Open Source Licenses

=====

This product may include a number of subcomponents with separate
copyright notices and license terms. Your use of the source code for

these subcomponents is subject to the terms and conditions of the subcomponent's license, as noted in the LICENSE-<subcomponent>.md files.

[[contributors]]

== Contributors

Browse the {junit5-repo}/graphs/contributors[current list of contributors] directly on GitHub.

1.355 slf4j-api-module 1.7.28

1.355.1 Available under license :

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

1.356 jackson-jaxrs-base 2.13.2

1.356.1 Available under license :

This copy of Jackson JSON processor databind module is licensed under the Apache (Software) License, version 2.0 ("the License").

See the License for details about distribution rights, and the specific rights regarding derivate works.

You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

1.357 jersey's-jersey 2.34

1.357.1 Available under license :

Notice for Jersey Core Server module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

org.objectweb.asm Version 8.0

* License: Modified BSD (<http://asm.objectweb.org/license.html>)

* Copyright: (c) 2000-2011 INRIA, France Telecom. All rights reserved.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Copyright (c) 2018 Oracle and/or its affiliates. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions

are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of the Eclipse Foundation, Inc. nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Notice for Jersey Core Common module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Google Guava Version 18.0

- * License: Apache License, 2.0

- * Copyright: (C) 2009 The Guava Authors

JSR-166 Extension - JEP 266

- * License: Creative Commons 1.0 (CC0)

- * No copyright

- * Written by Doug Lea with assistance from members of JCP JSR-166

- * Expert Group and released to the public domain, as explained at

- * <http://creativecommons.org/publicdomain/zero/1.0/>

Notice for Jersey Json Jackson module

This content is produced and maintained by the Eclipse Jersey project.

- * <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Jackson JAX-RS Providers version 2.10.1

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

/*

* Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.

*

* This program and the accompanying materials are made available under the

* terms of the Eclipse Public License v. 2.0, which is available at

* <http://www.eclipse.org/legal/epl-2.0>.

*

* This Source Code may also be made available under the following Secondary

* Licenses when the conditions for such availability set forth in the

* Eclipse Public License v. 2.0 are satisfied: GNU General Public License,

* version 2 with the GNU Classpath Exception, which is available at

* <https://www.gnu.org/software/classpath/license.html>.

*

* SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

*/

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Distribution License v. 1.0 which is available at <https://www.eclipse.org/org/documents/edl-v10.php>.

SPDX-License-Identifier: BSD-3-Clause

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey/examples>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.9.9

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 7.2

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. <http://www.w3.org/Consortium/Legal/>

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

- * Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed

source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

* License MIT (<http://www.opensource.org/licenses/mit-license.php>)

* Project: <http://angularjs.org>

* Copyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

* License: all the source code provided by AOP Alliance is Public Domain.

* Project: <http://aopalliance.sourceforge.net>

* Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

* License: Apache License, 2.0

* Project: <http://beanvalidation.org/1.1/>

* Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors

* by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

* License: Apache License, 2.0

* Project: <https://beanvalidation.org/>

* Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

* License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)

* Project: <http://getbootstrap.com>

* Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

- * License: Apache License, 2.0
- * Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.
- * Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

- * License: W3C License
- * Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.
<http://www.w3.org/Consortium/Legal/>
- # Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

- i) changes to the Program, and
- ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity. Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual

property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to

defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this

Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original

authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

- a) You must cause the modified files to carry prominent notices stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be

distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not

responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any

later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.

Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library

General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

/*

* Copyright (c) YYYY Oracle and/or its affiliates. All rights reserved.

*

* This program and the accompanying materials are made available under the
* terms of the Eclipse Distribution License v. 1.0, which is available at
* <http://www.eclipse.org/org/documents/edl-v10.php>.

*

* SPDX-License-Identifier: BSD-3-Clause

*/

Notice for Jersey Bean Validation module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms

of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Hibernate Validator CDI, 6.1.2.Final

* License: Apache License, 2.0

* Project: <https://beanvalidation.org/>

* Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

1.358 micronaut-views 3.2.0

1.358.1 Available under license :

No license file was found, but licenses were detected in source scan.

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

distributed under the License is distributed on an "AS IS" BASIS,

Found in path(s):

* `/opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/META-INF/native-image/io.micronaut.views.model.security/native-image.properties`

* `/opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/META-INF/native-image/io.micronaut.views.csp/native-image.properties`

No license file was found, but licenses were detected in source scan.

/*

* Copyright 2017-2020 original authors

*

* Licensed under the Apache License, Version 2.0 (the "License");

* you may not use this file except in compliance with the License.

* You may obtain a copy of the License at

*

* <https://www.apache.org/licenses/LICENSE-2.0>

*

- * Unless required by applicable law or agreed to in writing, software
- * distributed under the License is distributed on an "AS IS" BASIS,
- * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
- * See the License for the specific language governing permissions and
- * limitations under the License.
- */

Found in path(s):

- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/csp/package-info.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/View.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/ViewsConfigurationProperties.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/csp/CspConfiguration.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/csp/CspFilter.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/ModelAndView.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/ViewModelProcessor.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/ViewUtils.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/ViewsRenderer.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/security/SecurityViewModelProcessor.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/ViewsConfiguration.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/package-info.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/package-info.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/security/SecurityViewModelProcessorConfigurationProperties.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/exceptions/ViewRenderingException.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/exceptions/package-info.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/security/package-info.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/exceptions/ViewNotFoundException.java
- * /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-jar/io/micronaut/views/model/security/SecurityViewModelProcessorConfiguration.java

No license file was found, but licenses were detected in source scan.

```
/*
 * Copyright 2017-2021 original authors
 *
 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at
 *
 * https://www.apache.org/licenses/LICENSE-2.0
 *
 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 */
```

Found in path(s):

```
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/ViewsFilter.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/ViewsResolver.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/DefaultViewsRendererLocator.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/ViewsModelDecorator.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/DefaultViewsResolver.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/DefaultViewsModelDecorator.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/ViewsRendererKey.java
* /opt/cola/permits/1331474137_1653511933.527574/0/micronaut-views-core-3-2-0-sources-
jar/io/micronaut/views/ViewsRendererLocator.java
```

1.359 jersey-container-servlet 2.32

1.359.1 Available under license :

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Coyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity.

Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements

of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance

claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies
of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we

want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices

stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost

of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to

these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey

the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you

may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

1.360 dropwizard-utility-classes 2.0.18

1.360.1 Available under license :

Dropwizard

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

This product includes software developed by Coda Hale and Yammer, Inc.

Apache License
Version 2.0, January 2004
<http://www.apache.org/licenses/>

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

"License" shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

"Licensor" shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

"Legal Entity" shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, "control" means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

"You" (or "Your") shall mean an individual or Legal Entity exercising permissions granted by this License.

"Source" form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Object" form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

"Work" shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity

on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.
3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.
4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:
 - (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and
 - (b) You must cause any modified files to carry prominent notices stating that You changed the files; and
 - (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and
 - (d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one

of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions.

Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a

result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, with the fields enclosed by brackets "[]" replaced with your own identifying information. (Don't include the brackets!) The text should be enclosed in the appropriate comment syntax for the file format. We also recommend that a file or class name and description of purpose be included on the same "printed page" as the copyright notice for easier identification within third-party archives.

Copyright 2010-2013 Coda Hale and Yammer, Inc., 2014-2020 Dropwizard Team

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.

1.361 jersey-media-json-jackson 2.31

1.361.1 Available under license :

Notice for Jersey Json Jackson module

This content is produced and maintained by the Eclipse Jersey project.

* <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

* <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Jackson JAX-RS Providers version 2.10.1

* License: Apache License, 2.0

* Project: <https://github.com/FasterXML/jackson-jaxrs-providers>

* Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

Notice for Jersey

This content is produced and maintained by the Eclipse Jersey project.

* Project home: <https://projects.eclipse.org/projects/ee4j.jersey>

Trademarks

Eclipse Jersey is a trademark of the Eclipse Foundation.

Copyright

All content is the property of the respective authors or their employers. For more information regarding authorship of content, please consult the listed source code repository logs.

Declared Project Licenses

This program and the accompanying materials are made available under the terms of the Eclipse Public License v. 2.0 which is available at <http://www.eclipse.org/legal/epl-2.0>. This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License v. 2.0 are satisfied: GNU General Public License, version 2 with the GNU Classpath Exception which is available at <https://www.gnu.org/software/classpath/license.html>.

SPDX-License-Identifier: EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0

Source Code

The project maintains the following source code repositories:

- * <https://github.com/eclipse-ee4j/jersey>

Third-party Content

Angular JS, v1.6.6

- * License MIT (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://angularjs.org>
- * Coyright: (c) 2010-2017 Google, Inc.

aopalliance Version 1

- * License: all the source code provided by AOP Alliance is Public Domain.
- * Project: <http://aopalliance.sourceforge.net>
- * Copyright: Material in the public domain is not protected by copyright

Bean Validation API 2.0.2

- * License: Apache License, 2.0
- * Project: <http://beanvalidation.org/1.1/>
- * Copyright: 2009, Red Hat, Inc. and/or its affiliates, and individual contributors
- * by the @authors tag.

Hibernate Validator CDI, 6.1.2.Final

- * License: Apache License, 2.0
- * Project: <https://beanvalidation.org/>
- * Repackaged in `org.glassfish.jersey.server.validation.internal.hibernate`

Bootstrap v3.3.7

- * License: MIT license (<https://github.com/twbs/bootstrap/blob/master/LICENSE>)
- * Project: <http://getbootstrap.com>
- * Copyright: 2011-2016 Twitter, Inc

Google Guava Version 18.0

- * License: Apache License, 2.0
- * Copyright (C) 2009 The Guava Authors

javax.inject Version: 1

- * License: Apache License, 2.0
- * Copyright (C) 2009 The JSR-330 Expert Group

Javassist Version 3.25.0-GA

- * License: Apache License, 2.0
- * Project: <http://www.javassist.org/>
- * Copyright (C) 1999- Shigeru Chiba. All Rights Reserved.

Jackson JAX-RS Providers Version 2.10.1

- * License: Apache License, 2.0
- * Project: <https://github.com/FasterXML/jackson-jaxrs-providers>
- * Copyright: (c) 2009-2011 FasterXML, LLC. All rights reserved unless otherwise indicated.

jQuery v1.12.4

- * License: jquery.org/license
- * Project: jquery.org
- * Copyright: (c) jQuery Foundation

jQuery Barcode plugin 0.3

- * License: MIT & GPL (<http://www.opensource.org/licenses/mit-license.php> & <http://www.gnu.org/licenses/gpl.html>)
- * Project: <http://www.pasella.it/projects/jquery/barcode>
- * Copyright: (c) 2009 Antonello Pasella antonello.pasella@gmail.com

JSR-166 Extension - JEP 266

- * License: CC0
- * No copyright
- * Written by Doug Lea with assistance from members of JCP JSR-166 Expert Group and released to the public domain, as explained at <http://creativecommons.org/publicdomain/zero/1.0/>

KineticJS, v4.7.1

- * License: MIT license (<http://www.opensource.org/licenses/mit-license.php>)
- * Project: <http://www.kineticjs.com>, <https://github.com/ericdrowell/KineticJS>
- * Copyright: Eric Rowell

org.objectweb.asm Version 8.0

- * License: Modified BSD (<http://asm.objectweb.org/license.html>)
- * Copyright (c) 2000-2011 INRIA, France Telecom. All rights reserved.

org.osgi.core version 6.0.0

- * License: Apache License, 2.0
- * Copyright (c) OSGi Alliance (2005, 2008). All Rights Reserved.

org.glassfish.jersey.server.internal.monitoring.core

* License: Apache License, 2.0

* Copyright (c) 2015-2018 Oracle and/or its affiliates. All rights reserved.

* Copyright 2010-2013 Coda Hale and Yammer, Inc.

W3.org documents

* License: W3C License

* Copyright: Copyright (c) 1994-2001 World Wide Web Consortium, (Massachusetts Institute of Technology, Institut National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved.

<http://www.w3.org/Consortium/Legal/>

Eclipse Public License - v 2.0

THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS ECLIPSE PUBLIC LICENSE ("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.

1. DEFINITIONS

"Contribution" means:

a) in the case of the initial Contributor, the initial content Distributed under this Agreement, and

b) in the case of each subsequent Contributor:

i) changes to the Program, and

ii) additions to the Program;

where such changes and/or additions to the Program originate from and are Distributed by that particular Contributor. A Contribution "originates" from a Contributor if it was added to the Program by such Contributor itself or anyone acting on such Contributor's behalf. Contributions do not include changes or additions to the Program that are not Modified Works.

"Contributor" means any person or entity that Distributes the Program.

"Licensed Patents" mean patent claims licensable by a Contributor which are necessarily infringed by the use or sale of its Contribution alone or when combined with the Program.

"Program" means the Contributions Distributed in accordance with this Agreement.

"Recipient" means anyone who receives the Program under this Agreement or any Secondary License (as applicable), including Contributors.

"Derivative Works" shall mean any work, whether in Source Code or other form, that is based on (or derived from) the Program and for which the

editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship.

"Modified Works" shall mean any work in Source Code or other form that results from an addition to, deletion from, or modification of the contents of the Program, including, for purposes of clarity any new file in Source Code form that contains any contents of the Program. Modified Works shall not include works that contain only declarations, interfaces, types, classes, structures, or files of the Program solely in each case in order to link to, bind by name, or subclass the Program or Modified Works thereof.

"Distribute" means the acts of a) distributing or b) making available in any manner that enables the transfer of a copy.

"Source Code" means the form of a Program preferred for making modifications, including but not limited to software source code, documentation source, and configuration files.

"Secondary License" means either the GNU General Public License, Version 2.0, or any later versions of that license, including any exceptions or additional permissions as identified by the initial Contributor.

2. GRANT OF RIGHTS

a) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, Distribute and sublicense the Contribution of such Contributor, if any, and such Derivative Works.

b) Subject to the terms of this Agreement, each Contributor hereby grants Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed Patents to make, use, sell, offer to sell, import and otherwise transfer the Contribution of such Contributor, if any, in Source Code or other form. This patent license shall apply to the combination of the Contribution and the Program if, at the time the Contribution is added by the Contributor, such addition of the Contribution causes such combination to be covered by the Licensed Patents. The patent license shall not apply to any other combinations which include the Contribution. No hardware per se is licensed hereunder.

c) Recipient understands that although each Contributor grants the licenses to its Contributions set forth herein, no assurances are provided by any Contributor that the Program does not infringe the patent or other intellectual property rights of any other entity.

Each Contributor disclaims any liability to Recipient for claims brought by any other entity based on infringement of intellectual property rights or otherwise. As a condition to exercising the rights and licenses granted hereunder, each Recipient hereby assumes sole responsibility to secure any other intellectual property rights needed, if any. For example, if a third party patent license is required to allow Recipient to Distribute the Program, it is Recipient's responsibility to acquire that license before distributing the Program.

d) Each Contributor represents that to its knowledge it has sufficient copyright rights in its Contribution, if any, to grant the copyright license set forth in this Agreement.

e) Notwithstanding the terms of any Secondary License, no Contributor makes additional grants to any Recipient (other than those set forth in this Agreement) as a result of such Recipient's receipt of the Program under the terms of a Secondary License (if permitted under the terms of Section 3).

3. REQUIREMENTS

3.1 If a Contributor Distributes the Program in any form, then:

a) the Program must also be made available as Source Code, in accordance with section 3.2, and the Contributor must accompany the Program with a statement that the Source Code for the Program is available under this Agreement, and informs Recipients how to obtain it in a reasonable manner on or through a medium customarily used for software exchange; and

b) the Contributor may Distribute the Program under a license different than this Agreement, provided that such license:

i) effectively disclaims on behalf of all other Contributors all warranties and conditions, express and implied, including warranties or conditions of title and non-infringement, and implied warranties or conditions of merchantability and fitness for a particular purpose;

ii) effectively excludes on behalf of all other Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) does not attempt to limit or alter the recipients' rights in the Source Code under section 3.2; and

iv) requires any subsequent distribution of the Program by any party to be under a license that satisfies the requirements

of this section 3.

3.2 When the Program is Distributed as Source Code:

- a) it must be made available under this Agreement, or if the Program (i) is combined with other material in a separate file or files made available under a Secondary License, and (ii) the initial Contributor attached to the Source Code the notice described in Exhibit A of this Agreement, then the Program may be made available under the terms of such Secondary Licenses, and
- b) a copy of this Agreement must be included with each copy of the Program.

3.3 Contributors may not remove or alter any copyright, patent, trademark, attribution notices, disclaimers of warranty, or limitations of liability ("notices") contained within the Program from any copy of the Program which they Distribute, provided that Contributors may add their own appropriate notices.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance

claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, AND TO THE EXTENT PERMITTED BY APPLICABLE LAW, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure such failure in a reasonable period of time after becoming aware of such noncompliance. If all Recipient's rights under this Agreement terminate, Recipient agrees to cease use and distribution of the Program as soon as reasonably practicable. However, Recipient's obligations under this Agreement and any licenses granted by Recipient relating to the Program shall continue and survive.

Everyone is permitted to copy and distribute copies of this Agreement, but in order to avoid inconsistency the Agreement is copyrighted and may only be modified in the following manner. The Agreement Steward reserves the right to publish new versions (including revisions) of this Agreement from time to time. No one other than the Agreement Steward has the right to modify this Agreement. The Eclipse Foundation is the initial Agreement Steward. The Eclipse Foundation may assign the responsibility to serve as the Agreement Steward to a suitable separate entity. Each new version of the Agreement will be given a distinguishing version number. The Program (including Contributions) may always be Distributed subject to the version of the Agreement under which it was received. In addition, after a new version of the Agreement is published, Contributor may elect to Distribute the Program (including its Contributions) under the new version.

Except as expressly stated in Sections 2(a) and 2(b) above, Recipient receives no rights or licenses to the intellectual property of any Contributor under this Agreement, whether expressly, by implication, estoppel or otherwise. All rights in the Program not expressly granted under this Agreement are reserved. Nothing in this Agreement is intended to be enforceable by any entity that is not a Contributor or Recipient. No third-party beneficiary rights are created under this Agreement.

Exhibit A - Form of Secondary Licenses Notice

"This Source Code may also be made available under the following Secondary Licenses when the conditions for such availability set forth in the Eclipse Public License, v. 2.0 are satisfied: {name license(s), version(s), and exceptions or additional permissions here}."

Simply including a copy of this Agreement, including this Exhibit A is not sufficient to license the Source Code under Secondary Licenses.

If it is not possible or desirable to put the notice in a particular file, then You may include the notice in a location (such as a LICENSE file in a relevant directory) where a recipient would be likely to look for such a notice.

You may add additional accurate notices of copyright ownership.

The GNU General Public License (GPL) Version 2, June 1991

Copyright (C) 1989, 1991 Free Software Foundation, Inc.
51 Franklin Street, Fifth Floor
Boston, MA 02110-1335
USA

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

Preamble

The licenses for most software are designed to take away your freedom to share and change it. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change free software--to make sure the software is free for all its users. This General Public License applies to most of the Free Software Foundation's software and to any other program whose authors commit to using it. (Some other Free Software Foundation software is covered by the GNU Library General Public License instead.) You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for this service if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs; and that you know you can do these things.

To protect your rights, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. These restrictions translate to certain responsibilities for you if you distribute copies of the software, or if you modify it.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

We protect your rights with two steps: (1) copyright the software, and (2) offer you this license which gives you legal permission to copy, distribute and/or modify the software.

Also, for each author's protection and ours, we want to make certain that everyone understands that there is no warranty for this free software. If the software is modified by someone else and passed on, we

want its recipients to know that what they have is not the original, so that any problems introduced by others will not reflect on the original authors' reputations.

Finally, any free program is threatened constantly by software patents. We wish to avoid the danger that redistributors of a free program will individually obtain patent licenses, in effect making the program proprietary. To prevent this, we have made it clear that any patent must be licensed for everyone's free use or not licensed at all.

The precise terms and conditions for copying, distribution and modification follow.

TERMS AND CONDITIONS FOR COPYING, DISTRIBUTION AND MODIFICATION

0. This License applies to any program or other work which contains a notice placed by the copyright holder saying it may be distributed under the terms of this General Public License. The "Program", below, refers to any such program or work, and a "work based on the Program" means either the Program or any derivative work under copyright law: that is to say, a work containing the Program or a portion of it, either verbatim or with modifications and/or translated into another language. (Hereinafter, translation is included without limitation in the term "modification".) Each licensee is addressed as "you".

Activities other than copying, distribution and modification are not covered by this License; they are outside its scope. The act of running the Program is not restricted, and the output from the Program is covered only if its contents constitute a work based on the Program (independent of having been made by running the Program). Whether that is true depends on what the Program does.

1. You may copy and distribute verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice and disclaimer of warranty; keep intact all the notices that refer to this License and to the absence of any warranty; and give any other recipients of the Program a copy of this License along with the Program.

You may charge a fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

2. You may modify your copy or copies of the Program or any portion of it, thus forming a work based on the Program, and copy and distribute such modifications or work under the terms of Section 1 above, provided that you also meet all of these conditions:

a) You must cause the modified files to carry prominent notices

stating that you changed the files and the date of any change.

b) You must cause any work that you distribute or publish, that in whole or in part contains or is derived from the Program or any part thereof, to be licensed as a whole at no charge to all third parties under the terms of this License.

c) If the modified program normally reads commands interactively when run, you must cause it, when started running for such interactive use in the most ordinary way, to print or display an announcement including an appropriate copyright notice and a notice that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License. (Exception: if the Program itself is interactive but does not normally print such an announcement, your work based on the Program is not required to print an announcement.)

These requirements apply to the modified work as a whole. If identifiable sections of that work are not derived from the Program, and can be reasonably considered independent and separate works in themselves, then this License, and its terms, do not apply to those sections when you distribute them as separate works. But when you distribute the same sections as part of a whole which is a work based on the Program, the distribution of the whole must be on the terms of this License, whose permissions for other licensees extend to the entire whole, and thus to each and every part regardless of who wrote it.

Thus, it is not the intent of this section to claim rights or contest your rights to work written entirely by you; rather, the intent is to exercise the right to control the distribution of derivative or collective works based on the Program.

In addition, mere aggregation of another work not based on the Program with the Program (or with a work based on the Program) on a volume of a storage or distribution medium does not bring the other work under the scope of this License.

3. You may copy and distribute the Program (or a work based on it, under Section 2) in object code or executable form under the terms of Sections 1 and 2 above provided that you also do one of the following:

a) Accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

b) Accompany it with a written offer, valid for at least three years, to give any third party, for a charge no more than your cost

of physically performing source distribution, a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Sections 1 and 2 above on a medium customarily used for software interchange; or,

c) Accompany it with the information you received as to the offer to distribute corresponding source code. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form with such an offer, in accord with Subsection b above.)

The source code for a work means the preferred form of the work for making modifications to it. For an executable work, complete source code means all the source code for all modules it contains, plus any associated interface definition files, plus the scripts used to control compilation and installation of the executable. However, as a special exception, the source code distributed need not include anything that is normally distributed (in either source or binary form) with the major components (compiler, kernel, and so on) of the operating system on which the executable runs, unless that component itself accompanies the executable.

If distribution of executable or object code is made by offering access to copy from a designated place, then offering equivalent access to copy the source code from the same place counts as distribution of the source code, even though third parties are not compelled to copy the source along with the object code.

4. You may not copy, modify, sublicense, or distribute the Program except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense or distribute the Program is void, and will automatically terminate your rights under this License. However, parties who have received copies, or rights, from you under this License will not have their licenses terminated so long as such parties remain in full compliance.

5. You are not required to accept this License, since you have not signed it. However, nothing else grants you permission to modify or distribute the Program or its derivative works. These actions are prohibited by law if you do not accept this License. Therefore, by modifying or distributing the Program (or any work based on the Program), you indicate your acceptance of this License to do so, and all its terms and conditions for copying, distributing or modifying the Program or works based on it.

6. Each time you redistribute the Program (or any work based on the Program), the recipient automatically receives a license from the original licensor to copy, distribute or modify the Program subject to

these terms and conditions. You may not impose any further restrictions on the recipients' exercise of the rights granted herein. You are not responsible for enforcing compliance by third parties to this License.

7. If, as a consequence of a court judgment or allegation of patent infringement or for any other reason (not limited to patent issues), conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot distribute so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not distribute the Program at all. For example, if a patent license would not permit royalty-free redistribution of the Program by all those who receive copies directly or indirectly through you, then the only way you could satisfy both it and this License would be to refrain entirely from distribution of the Program.

If any portion of this section is held invalid or unenforceable under any particular circumstance, the balance of the section is intended to apply and the section as a whole is intended to apply in other circumstances.

It is not the purpose of this section to induce you to infringe any patents or other property right claims or to contest validity of any such claims; this section has the sole purpose of protecting the integrity of the free software distribution system, which is implemented by public license practices. Many people have made generous contributions to the wide range of software distributed through that system in reliance on consistent application of that system; it is up to the author/donor to decide if he or she is willing to distribute software through any other system and a licensee cannot impose that choice.

This section is intended to make thoroughly clear what is believed to be a consequence of the rest of this License.

8. If the distribution and/or use of the Program is restricted in certain countries either by patents or by copyrighted interfaces, the original copyright holder who places the Program under this License may add an explicit geographical distribution limitation excluding those countries, so that distribution is permitted only in or among countries not thus excluded. In such case, this License incorporates the limitation as if written in the body of this License.

9. The Free Software Foundation may publish revised and/or new versions of the General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies a version number of this License which applies to it and "any later version", you have the option of following the terms and conditions either of that version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of this License, you may choose any version ever published by the Free Software Foundation.

10. If you wish to incorporate parts of the Program into other free programs whose distribution conditions are different, write to the author to ask for permission. For software which is copyrighted by the Free Software Foundation, write to the Free Software Foundation; we sometimes make exceptions for this. Our decision will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software generally.

NO WARRANTY

11. BECAUSE THE PROGRAM IS LICENSED FREE OF CHARGE, THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

12. IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MAY MODIFY AND/OR REDISTRIBUTE THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

END OF TERMS AND CONDITIONS

How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively convey

the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

One line to give the program's name and a brief idea of what it does.
Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1335 USA

Also add information on how to contact you by electronic and paper mail.

If the program is interactive, make it output a short notice like this when it starts in an interactive mode:

```
Gnomovision version 69, Copyright (C) year name of author
Gnomovision comes with ABSOLUTELY NO WARRANTY; for details type
`show w'. This is free software, and you are welcome to redistribute
it under certain conditions; type `show c' for details.
```

The hypothetical commands `show w' and `show c' should show the appropriate parts of the General Public License. Of course, the commands you use may be called something other than `show w' and `show c'; they could even be mouse-clicks or menu items--whatever suits your program.

You should also get your employer (if you work as a programmer) or your school, if any, to sign a "copyright disclaimer" for the program, if necessary. Here is a sample; alter the names:

```
Yoyodyne, Inc., hereby disclaims all copyright interest in the
program `Gnomovision' (which makes passes at compilers) written by
James Hacker.
```

```
signature of Ty Coon, 1 April 1989
Ty Coon, President of Vice
```

This General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you

may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Library General Public License instead of this License.

CLASSPATH EXCEPTION

Linking this library statically or dynamically with other modules is making a combined work based on this library. Thus, the terms and conditions of the GNU General Public License version 2 cover the whole combination.

As a special exception, the copyright holders of this library give you permission to link this library with independent modules to produce an executable, regardless of the license terms of these independent modules, and to copy and distribute the resulting executable under terms of your choice, provided that you also meet, for each linked independent module, the terms and conditions of the license of that module. An independent module is a module which is not derived from or based on this library. If you modify this library, you may extend this exception to your version of the library, but you are not obligated to do so. If you do not wish to do so, delete this exception statement from your version.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

©2022 Cisco Systems, Inc. All rights reserved.